

50376
1992
188

61592

50376
1992
188

n° d'ordre : 925

THESE
présentée à
**L'UNIVERSITE DES SCIENCES ET TECHNIQUES DE
LILLE I**

pour obtenir le titre de
DOCTEUR D'UNIVERSITE

spécialité Productique:
automatique et informatique industrielle

par

My-Anh DAO - GARNOT
ingénieur I.D.N.



**CONTRIBUTION AU CALCUL DE TRAJECTOIRES
OPTIMALES PAR LA METHODE DE QUASILINEARISATION
MISE EN ŒUVRE ET APPLICATION EN AERONAUTIQUE**

Soutenue le 9 Juin 1992 devant le jury d'examen

MM.	P. Borne	Président, rapporteur
	M. Kinnaert	Rapporteur
	C. Brie	Examineur
M ^{me}	G. Dauphin -Tanguy	Examineur
MM.	H.T. Huynh	Examineur
	M. Staroswiecki	Examineur

thèse codirigée par

M^{me} Dauphin-Tanguy, Professeur à l'Ecole Centrale de Lille,
M^f Huynh, Chef de subdivision à l'O.N.E.R.A. / Salon de provence.

à mon grand père et à mon oncle cậ ba
à mes parents pour leur aide et leur soutien
à Florent et Sylvain

AVANT PROPOS

Nous tenons à exprimer notre profonde gratitude à Madame G. Dauphin-Tanguy pour les conseils et les encouragements constants prodigués au cours de cette thèse accomplie à l'Office National d'Etudes et de Recherches Aérospatiales. Nous traduisons notre respectueuse reconnaissance à Monsieur H.T. Huynh pour nous avoir accueillie dans son équipe et pour avoir codirigé ces travaux avec beaucoup d'attention.

Nous sommes très honorée de la présence de Monsieur le Professeur P. Borne qui a accepté de présider ce jury et de porter un jugement cette thèse.

Nous avons également l'honneur de témoigner de notre reconnaissance à Monsieur le Professeur M. Kinnaert pour avoir accepté de juger ce travail.

Nous remercions vivement Messieurs les Professeurs C. Brie et M. Staroswiecki pour leur participation à ce jury.

Nous tenons à remercier également Monsieur C. Aumasson pour les entretiens très constructifs qu'il nous a accordés.

Enfin c'est avec un très grand plaisir que nous remercions tous ceux qui nous ont conseillée et soutenue au cours de ce long travail effectué à l'Office. En particulier, nous témoignons de notre profonde reconnaissance à toutes les personnes du service pour leur accueil amical et leur grande disponibilité.

TABLE DES MATIERES

AVANT PROPOS	5
TABLE DES MATIERES	7
INTRODUCTION GENERALE	15
NOTATIONS	21
CHAPITRE 1 : LE PROBLEME DE LA COMMANDE OPTIMALE	23
INTRODUCTION	25
1.RESULTATS GENERAUX	27
1.1.Formulation du problème	27
1.2.Théorie du calcul de variations	29
1.2.1.Conditions du premier ordre	29
1.2.2.Conditions du second ordre	33
1.3.Principe de PONTRYAGIN: une condition nécessaire d'optimalité	34
1.4.Equation de HAMILTON-BELLMAN-JACOBI: une condition suffisante d'optimalité	35
1.5.Traitement des contraintes	38
1.5.1.Contraintes sur la commande	39
1.5.2.Contraintes sur l'état	40
1.5.3.Contraintes sur l'état et la commande	41
1.5.4.Contraintes intégrales	42
2.METHODES NUMERIQUES D'OPTIMISATION	45
2.1.Méthodes du gradient	46
2.1.1. Méthodes du premier ordre	46
2.1.2. Méthodes du second ordre	50

2.2.Méthodes des extrémaux voisins et méthodes de tir	53
2.2.1. Méthodes des extrémaux voisins	53
2.2.2. Méthode de tir direct	54
2.2.3. Méthode de tir multiple	55
2.3.Méthodes basées sur des linéarisations successives	56
2.3.1. Méthode de quasilinearisation	56
2.3.2. Pseudolinéarisation et linéarisation par difféomorphisme.	57
2.4.Calcul d'une commande par retour d'état	58
2.4.1. Programmation dynamique différentielle	58
2.4.2. Méthode utilisant les crochets de Lie	61
2.4.3. Solution approchée par développement en séries	62
2.5.Méthodes paramétriques	65
2.5.1. Paramétrisation de la commande	65
2.5.2. Paramétrisation de l'état, de l'adjoint et/ou de la commande	67
CONCLUSION: SYNTHESE COMPARATIVE DES METHODES	71
CHAPITRE 2 : METHODE DE QUASILINEARISATION	75
INTRODUCTION	77
1.PRINCIPE DE LA METHODE	79
1.1.méthode de résolution de $\vec{G}(\vec{z}) = 0$	79
1.2.Opérateur de Newton-Raphson généralisé	81
1.3.Application à la commande optimale	83
2.DEVELOPPEMENT D'UNE NOUVELLE VARIANTE	87
2.1.Présentation générale	87
2.2.Description d'une itération	88
2.2.1. Calcul de variation	89
Variations sur la dynamique	89
Variations sur les conditions initiales et finales	90

2.2.2. Résolution du système différentiel linéarisé	91
calcul de la matrice de transition	92
calcul direct des sensibilités	94
calcul par équation de RICCATI	95
2.2.3. Domaine de validité et généralisation	97
validité de la méthode	97
problèmes à temps final libre	98
contraintes intégrales	100
2.3. Gestion des itérations	102
2.3.1. Etude de la convergence	102
2.3.2. Calcul des matrices de contraction	105
2.4. Recherche d'une "bonne"initialisation	107
2.4.1. Initialisations par intégration	107
2.4.2. Initialisation du vecteur adjoint	108
3. MISE EN OEUVRE	111
3.1. Définition du guidage	112
3.2. Description du programme	115
3.2.1. structure du programme	115
organigramme	116
3.2.2. description des modules	117
3.2.3. choix des données et des méthodes	118
3.3. Exemples numériques	120
3.3.1. Exemple préliminaire	120
3.3.2. Décollage du L.E.M. lunaire	126
3.3.3. Exemple de Yeo	131
CONCLUSION	133
CHAPITRE 3 : EXEMPLES D'APPLICATION	135
INTRODUCTION	137
1. PROBLEME D'INTERCEPTION	139
1.1. Description du modèle	139
1.1.1. Définition des repères, du vecteur d'état et du vecteur des commandes	139
1.1.2. Equations du mouvement	140

Modèle de propulsion	141
Modèle aérodynamique	141
Equations d'état	142
1.1.3. Domaine de vol	142
1.2. Formulation générale du problème d'optimisation	143
1.2.1. Equations du problème	143
1.2.2. Résolution analytique	145
1.2.3. Résultats numériques	147
2.EVALUATION DES PERFORMANCES D'UN HELICOPTERE POUR UNE MANOEUVRE DE SAUT	151
2.1. Description du modèle physique	151
2.1.1. Définition des repères, du vecteur d'état et de commandes	151
Etat de l'hélicoptère	152
Commandes	153
2.1.2. Equations du mouvement dans le plan vertical	153
Dynamique du centre de gravité	153
Modèle aérodynamique	154
Bilan de puissance	156
2.1.3. Définition du domaine de vol	157
2.1.4. Simplification du modèle	158
Traitement de la contrainte sur l'état	158
Simplification des équations d'état bornes sur les commandes	158 159
2.2. Evaluation du gain en performance par le contrôle actif moteur	159
2.2.1. Problème d'optimisation	159
2.2.2. résolution analytique	160
analyse de l'incidence optimale	162
2.2.3. Résultats numériques	162
CONCLUSION	165
CONCLUSION GENERALE	167

PLANCHES

Planche 1 : Courbes obtenues pour l'exemple de Yeo

Planche 2 : facteurs de charge limite

Planche 3 : coefficients de trainée

Planche 4: coefficients de portance

Planche 5 : courbes de poussée maximale

Planche 6 : Evolution du cout en fonction de itérations pour le gradient projeté

Planches 7 à 10 : Résultat du gradient projeté pour le cas sans contrôle moteur

Planches 11 à 14 : Résultat de la quasilinearisation

Planches 15 à 18 : Résultat du gradient projeté pour le cas avec contrôle moteur

Planches 19 à 22 : Résultat de la quasilinearisation

REFERENCES BIBLIOGRAPHIQUES

B1

ANNEXES

A1

Annexe 1 : variation de la condition de transversalité

A3

Annexe 2 : solution analytique du problème linéaire quadratique

A7

Annexe 3 : Matrices de changement de repères
Application du théorème de la dynamique

A11

Annexe 4 : changement de variables

A13

Annexe 5 : commande par perturbations singulières

A15

Annexe 6 : calcul des dérivées partielles du système hélicoptère

A17

Annexe 7 : commandes en vol équilibré

A19

INTRODUCTION GENERALE

L'optimisation des systèmes dynamiques suscite depuis longtemps beaucoup d'intérêt tant dans le domaine économique que dans le domaine des sciences de l'ingénieur. Dès les années 50, le problème des systèmes régis par des équations différentielles a reçu l'attention de nombreux chercheurs et les études théoriques développées à ce jour sont nombreuses. Les premiers résultats concernant les conditions nécessaires d'optimalité sont donnés par la théorie classique du calcul des variations. En 1962, Pontryagin démontre de manière rigoureuse une condition nécessaire d'optimalité. Bellman énonce en 1957 un principe d'optimalité, donnant naissance à la programmation dynamique différentielle, et fournit des conditions suffisantes d'optimalité.

Du point de vue des applications, un grand intérêt est porté sur le calcul d'une commande optimale non linéaire en boucle fermée. En particulier dans le domaine aérospatial où les systèmes étudiés sont complexes et très non linéaires, on cherche à embarquer des lois de commandes optimales pour le guidage temps-réel. Cependant, en dehors de la théorie de Kalman qui permet d'exprimer la commande optimale en fonction de l'état moyennant la résolution d'une équation matricielle différentielle de Riccati, la commande en boucle fermée reste difficile à calculer. En effet il n'existe pas, dans le cas non linéaire, de résolution analytique simple de ce problème. Le calcul de la commande optimale est, de ce fait, effectué de façon itérative par des algorithmes numériques.

Ces algorithmes développés dans les années 70 sont en général destinés à l'obtention off-line de trajectoires optimales. Pourtant depuis les années 80, les progrès techniques des calculateurs permettent d'envisager l'utilisation d'une méthode boucle ouverte pour le guidage temps réel. Afin d'évaluer les possibilités et les limites d'une telle application, il est nécessaire de faire le choix d'une méthode à convergence rapide. Cette thèse, effectuée à l'O.N.E.R.A., Office National d'Etudes et de Recherches Aérospatiales, a pour but d'une part, de recenser les méthodes d'optimisation existant dans la littérature et d'autre part, de mettre en œuvre et de valider sur des exemples aéronautiques une méthode à convergence rapide: la quasilinearisation.

Cette méthode a été retenue car elle permet de résoudre la plupart des problèmes d'optimisation de manœuvres rencontrées dans le domaine aérospatial, avec la possibilité de prendre en compte des contraintes non linéaires portant à la

fois sur les commandes (limitation de braquages de gouvernes) et sur l'état (limitation de facteur de charge, domaine de vol). Elle présente, de plus, par rapport aux méthodes numériques du premier ordre, de type gradient projeté, l'avantage essentiel d'une convergence rapide. Néanmoins, son emploi en pratique reste jusqu'à présent limité du fait de son faible domaine de convergence. La contribution essentielle de cette thèse consistera d'une part à proposer des améliorations permettant d'accroître la facilité d'utilisation de cette méthode notamment en ce qui concerne l'initialisation et d'autre part à les valider sur des exemples aéronautiques aussi réalistes que possible.

Après avoir formulé le problème d'optimisation et précisé les hypothèses, nous rappelons les résultats théoriques généraux. Dans ce chapitre 1, nous effectuons un bilan détaillé des méthodes de calcul de commande optimale trouvées dans la littérature. Nous décrivons le principe de chaque méthode et nous analysons leurs caractéristiques ainsi que leur efficacité dans la mise en œuvre (encombrement mémoire, volume de calculs, vitesse de convergence, complexité, champs d'application...). Enfin une synthèse comparative de ces méthodes est présentée. Le but de cette synthèse sera de guider l'ingénieur dans le choix de la méthode la plus appropriée pour résoudre un problème d'optimisation donné.

Le chapitre 2 consiste à présenter le principe de la méthode de quasilinearisation. Après une présentation de l'état de l'art, nous développons en détail une variante de la méthode. Basée sur le principe de la méthode de Newton, cette méthode choisie pour sa rapidité de convergence, a un domaine de convergence relativement faible. En effet, elle utilise la linéarisation des équations d'optimalité, ce qui implique le calcul de variations du second ordre et une grande sensibilité à l'initialisation. Nous analysons les possibilités d'améliorer l'algorithme. Une première approche consiste à modifier la gestion des itérations. Nous proposons dans cette optique une nouvelle extension de la méthode qui consiste à modifier les variations de l'état, de l'adjoint et du multiplicateur de Lagrange en fonction l'erreur relative des fonctions par rapport à une optimale estimée. Une seconde approche consiste à trouver une initialisation proche de l'optimum, pour cela, nous recensons les différentes possibilités d'initialisation. Un programme général a été développé et mis en œuvre sur des cas simples afin de tester son comportement .

Dans le chapitre 3, nous étudions deux exemples aéronautiques sur lesquels nous appliquons la méthode. Le premier exemple est un problème d'interception

dans le plan horizontal pour un avion de combat. Cet exemple nous permet de comparer la vitesse de convergence de l'algorithme par rapport à une méthode du premier ordre, le gradient projeté, en fin de convergence. Nous initialisons les algorithmes avec la solution des perturbations singulières. Cette solution donne, en effet, une bonne approximation de la solution optimale, ce qui permet de se placer dans le cas d'une convergence finale. Le second exemple a pour but l'évaluation du gain de performances en manœuvre pour un hélicoptère type grâce à un accroissement de vitesse de rotation du rotor. Nous étudions les résultats obtenus par le gradient projeté et la quasilinéarisation dans deux cas: pour une vitesse de rotation du rotor fixée ou variable. L'algorithme développé est alors utilisé comme un outil qui nous permet de confirmer rapidement l'optimalité d'une trajectoire lorsque les méthodes du premier ordre ne donnent plus d'améliorations significatives.

NOTATIONS.

Un vecteur de \mathbb{R}^n sera représenté par
Ses composantes seront notées

$$\vec{x}$$

$$x_i \quad i=1,n$$

Une matrice $m \times n$ sera représentée par
Ses composantes seront notées
où i indiquera par convention l'indice de ligne
et j l'indice de colonne

$$\mathbf{A}$$

$$a_{ij} \quad i=1,m \quad j=1,n$$

Soit une fonction scalaire à variables vectorielles

$$f : \mathbb{R}^n \text{ -----} \rightarrow \mathbb{R}$$

$$: \vec{x} \text{ -----} \rightarrow f(\vec{x})$$

Le gradient transposé en \vec{x} de f sera représenté par
vecteur ligne des dérivées partielles

$$\frac{\partial f}{\partial \vec{x}}$$

$$\left(\frac{\partial f}{\partial x_1} \cdots \frac{\partial f}{\partial x_n} \right) = (\text{grad } f)^T$$

La dérivée seconde de f sera représentée par

$$\frac{\partial^2 f}{\partial \vec{x}^2}$$

$$\begin{pmatrix} \frac{\partial^2 f}{\partial x_1 \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \cdots & \cdots & \cdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_n \partial x_n} \end{pmatrix}$$

matrice $n \times n$ des dérivées partielles

Soit une fonction vectorielle \vec{f}

$$\vec{f} : \mathbb{R}^n \text{ -----} \rightarrow \mathbb{R}^p$$

$$: \vec{x} \text{ -----} \rightarrow \vec{f}(\vec{x})$$

La dérivée de \vec{f} sera notée

$$\frac{\partial \vec{f}}{\partial \vec{x}}$$

$$\begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \cdots & \cdots & \cdots \\ \frac{\partial f_p}{\partial x_1} & \cdots & \frac{\partial f_p}{\partial x_n} \end{pmatrix}$$

matrice $p \times n$ des dérivées partielles

Notations particulières

vecteur état	$\vec{x} \in \mathbb{R}^{n_x}$
vecteur de commandes	$\vec{u} \in \mathbb{R}^{n_u}$
vecteur adjoint	$\vec{\lambda} \in \mathbb{R}^{n_x}$
multiplicateur de Lagrange	$\vec{v} \in \mathbb{R}^{n_{cf}}$
critère d'optimisation	$J \in \mathbb{R}$
terme terminal du critère d'optimisation	$\varphi \in \mathbb{R}$
terme intégral du critère d'optimisation	$L \in \mathbb{R}$
contraintes finales égalité	$\vec{\psi} \in \mathbb{R}^{n_{cf}}$
Matrice de transition d'un système linéaire $n \times n$: défini par:	$\mathbf{T}(t, t_0) \in \mathbb{R}^n$ $\dot{\mathbf{T}}(t, t_0) = \mathbf{A}(t) \cdot \mathbf{T}(t, t_0)$ $\mathbf{T}(t_0, t_0) = \mathbf{I}_n$

CHAPITRE 1

LE PROBLEME DE LA COMMANDE OPTIMALE

CHAPITRE 1

LE PROBLEME DE LA COMMANDE OPTIMALE

INTRODUCTION

La première partie de ce chapitre est consacrée à la formulation du problème et à la présentation des résultats théoriques généraux.

Nous précisons, dans un premier temps, les données du problème et les hypothèses envisagées. Dans un second temps, nous rappelons les résultats théoriques fondamentaux relatifs au calcul de la commande optimale des systèmes dynamiques. Trois approches principales sont décrites :

- la théorie du calcul de variation,
- le principe de PONTRYAGIN,
- l'équation de HAMILTON-BELLMAN-JACOBI.

Ces résultats sont la base de toutes les méthodes de résolution présentées par la suite. Enfin, nous exposons les résultats concernant l'optimisation des systèmes soumis à ces contraintes.

La seconde partie recense les principales méthodes de résolution développées à partir des résultats théoriques. Nous présentons tout d'abord le principe de chaque méthode. Ces méthodes sont nombreuses et diverses, nous les avons classées en quatre groupes:

- les méthodes directes du premier ou du second ordre
- les méthodes indirectes
- les méthodes de calcul par retour d'état
- les méthodes paramétriques.

Aucune méthode n'étant universelle, la méthode la plus appropriée pour le problème étudié pourra être différente selon le but recherché. Nous effectuons dans une dernière partie une comparaison de ces méthodes. Le but de cette synthèse est de donner une vue générale de tous les types de résolution possibles et d'indiquer leurs caractéristiques afin de guider l'ingénieur pour le choix d'une méthode. Nous examinons les critères suivants:

- simplicité de formulation
- domaine de convergence
- vitesse de convergence
- contraintes traitées
- capacité mémoire requise
- possibilité de calcul en boucle fermée
- champs d'application de la méthode.

1.RESULTATS GENERAUX

Nous précisons ici, les données du problème, les hypothèses envisagées ainsi que les résultats théoriques fondamentaux. Les notations définies précédemment nous permettent une présentation unifiée des différentielles de fonctions scalaires et vectorielles.

Nous présenterons à la fin de cette partie, certains résultats concernant le traitement des contraintes.

1.1.Formulation du problème

Nous considérons les systèmes dynamiques non linéaires déterministes définis par leur représentation d'état. L'évolution de ces systèmes sera donnée par le système différentiel:

$$\dot{\vec{x}} = \vec{f}(\vec{x}(t), \vec{u}(t), t) \quad (1.1)$$

où \vec{f} est une fonction de classe C^2 des variables \vec{x} et \vec{u} , qui sont respectivement, le vecteur état de dimension n_x et le vecteur commande de dimension n_u et où t est une variable scalaire qui représente le temps.

Définition : nous appellerons trajectoire associée à une commande \vec{u} , la fonction $\vec{x}(t)$ obtenue avec cette commande.

Soit un ensemble Ω de commandes admissibles $\vec{u}(t)$. Par exemple, Ω pourra être une partie de l'ensemble des fonctions continues par morceaux.

Soient les conditions initiales et finales suivantes:

$$\vec{x}(t_0) = \vec{x}_0 \quad (1.2)$$

t_0 instant initial donné

$$\vec{x}(t_f) \in S(t_f) \quad (1.3)$$

$$S(t) = \{ \vec{x}(t) \text{ tels que } \vec{\psi}(\vec{x}(t), t) = 0 \}$$

où t_f est un instant final donné et $\vec{\psi}(\vec{x}(t), t)$ une fonction vectorielle différentiable de dimension n_{cf} qui représente les contraintes finales.

Considérons un indice de performance défini par :

$$J = \int_{t_0}^{t_f} L(\vec{x}(t), \vec{u}(t), t) dt + \varphi(\vec{x}(t_f), t_f) \quad (1.4)$$

où $\varphi(\vec{x}, t)$ est une fonction continuellement différentiable en \vec{x} et t et L une fonction de classe C^2 de ses variables.

Le problème de la commande optimale sera le suivant :

Soient donnés

un système dynamique $\dot{\vec{x}} = \vec{F}(\vec{x}(t), \vec{u}(t), t)$

des conditions initiales $\vec{x}(t_0) = \vec{x}_0$

un ensemble Ω de commandes admissibles

un ensemble $S(t)$ de contraintes finales,

trouver une commande $\vec{u}(t)$ telle que (P1)

$\vec{u}(t) \in \Omega$

\vec{x} évolue de \vec{x}_0 à $\vec{x}(t_f) \in S(t_f)$

le critère J soit minimum.

Ce problème est le problème classique de Bolza dans la théorie du calcul de variations. Il s'agira du problème de Lagrange si le terme final est nul et de Mayer si l'intégrande est nul. En fait toutes ces formes de l'indice de performance peuvent être ramenées à un critère final du type de Mayer. En effet, si nous définissons un état supplémentaire x_{n_x+1} :

$$\begin{aligned} \dot{x}_{n_x+1} &= L(\vec{x}(t), \vec{u}(t)) \\ x_{n_x+1}(t_0) &= 0 \end{aligned}$$

Le critère devient alors :

$$J = x_{n_x+1}(t_f) + \varphi(\vec{x}(t_f), t_f)$$

Hypothèses :

H1) Nous supposons que le système est localement complètement commandable sur $[t_0; t_f]$ le long de toute trajectoire correspondant à une commande admissible. Cette hypothèse se traduit par :

$$\int_{t_0}^t \mathbf{T}(t, \tau) \cdot \frac{\partial \vec{f}}{\partial \vec{u}} \cdot \frac{\partial \vec{f}^T}{\partial \vec{u}} \cdot \mathbf{T}(t, \tau)^T \cdot dt \text{ inversible}$$

pour tout $t \in [t_0; t_f]$

\mathbf{T} est la matrice de transition du système linéarisé

$$\overset{\circ}{\mathbf{T}}(t, t_0) = \frac{\partial \vec{f}}{\partial \vec{x}} \cdot \mathbf{T}(t, t_0)$$

$$\mathbf{T}(t_0, t_0) = \mathbf{I}_{n_x} \text{ (matrice identité de } \mathbb{R}^{n_x}\text{)}$$

H2) Nous supposons par ailleurs qu'il existe une solution au problème, c'est à dire que pour J^* minimum il existe une commande et une trajectoire qui correspondent à cette valeur de J .

1.2. Théorie du calcul de variations

1.2.1. Conditions du premier ordre

Définissons la fonction hamiltonienne H et les contraintes finales augmentées:

$$H = L + \vec{\lambda}^T \cdot \vec{f} \tag{1.5}$$

$$\Phi = \varphi + \vec{v}^T \cdot \vec{\psi} \tag{1.6}$$

où \vec{v} est le vecteur constant des multiplicateurs de Lagrange et $\vec{\lambda}(t)$, un vecteur appelé vecteur adjoint que nous définirons ultérieurement.

Considérons l'indice de performance augmenté :

$$\bar{J} = \int_{t_0}^{t_f} (H - \vec{\lambda}^T \cdot \dot{\vec{x}}) \cdot dt + \Phi (\vec{x}(t_f), \vec{v}, t_f) \tag{1.7}$$

et une commande nominale et sa trajectoire correspondante telles que :

$$\vec{\Psi}(\vec{x}(t_f), t_f) = 0$$

alors, comme $\vec{s} = \vec{f}$, J sera égal à l'indice augmenté et en développant l'intégrale par partie nous aurons :

$$\begin{aligned} J &= \bar{J} \quad \text{et} \\ \bar{J} &= \Phi(\vec{x}(t_f), \vec{v}, t_f) - \left(\vec{\lambda}(t_f)^T \vec{x}(t_f) - \vec{\lambda}(t_0)^T \vec{x}(t_0) \right) \\ &\quad + \int_{t_0}^{t_f} \left(H + \vec{\lambda}^T \vec{x} \right) dt \end{aligned} \quad (1.8)$$

Ainsi pour une variation "faible" de la commande autour de la nominale, la variation au premier ordre de J sera :

$$\begin{aligned} \delta J &= \int_{t_0}^{t_f} \left(\frac{\partial H}{\partial \vec{u}} \cdot \delta \vec{u} + \left(\frac{\partial H}{\partial \vec{x}} + \vec{\lambda}^T \right) \cdot \delta \vec{x} \right) dt \\ &\quad + \left(\frac{\partial \Phi}{\partial \vec{x}} - \vec{\lambda}^T \right) \cdot \delta \vec{x} \Big|_f \\ &\quad + \left(\vec{\lambda}^T \cdot \delta \vec{x} \right)_{t_0} \end{aligned} \quad (1.9)$$

et si nous définissons le vecteur adjoint

$$\vec{\lambda}^T = - \frac{\partial H}{\partial \vec{x}} \quad (1.10)$$

$$\vec{\lambda}^T(t_f) = \left(\frac{\partial \Phi}{\partial \vec{x}} \right)_{t_f} \quad (1.11)$$

la variation δJ deviendra

$$\delta J = \int_{t_0}^{t_f} \left(\frac{\partial H}{\partial \vec{u}} \cdot \delta \vec{u} \right) dt + \left(\vec{\lambda}^T(t_0) \cdot \delta \vec{x}(t_0) \right) \quad (1.12)$$

Par conséquent, si u^* est optimal, la variation de J au premier ordre est nulle et nous aurons pour des conditions initiales fixées, la condition nécessaire :

$$\frac{\partial H}{\partial \vec{u}} \cdot \delta \vec{u} = 0 \quad (1.13)$$

Définition : nous appellerons variation de commande admissible, une fonction $\delta \vec{u}(t)$ telle que

$$\forall t \in [t_0; t_f]$$

$$\vec{u}(t) + \delta \vec{u}(t) \in \Omega$$

Nous avons alors deux cas : soit la commande optimale est intérieure à Ω , toute variation sera une variation admissible, soit la commande se trouve sur la frontière (lorsque celle-ci n'est pas vide) et les variations admissibles ne seront pas quelconques.

Dans le cas où la commande optimale est intérieure à Ω , nous obtenons des conditions nécessaires d'optimalité données par les équations classiques d'Euler Lagrange

$$\frac{\partial H}{\partial \vec{x}} = \frac{\partial H}{\partial \vec{\lambda}} \quad \vec{x}(t_0) = \vec{x}_0 \quad (1.14)$$

$$\frac{\partial H}{\partial \vec{\lambda}} = - \frac{\partial H}{\partial \vec{x}} \quad \vec{\lambda}^T(t_f) = \left(\frac{\partial \Phi}{\partial \vec{x}} \right)_{t_f} \quad (1.15)$$

$$\frac{\partial H}{\partial \vec{u}} = 0 \quad (1.16)$$

Dans le cas où la commande optimale se trouve sur la frontière de Ω , la condition 1.17 est remplacée par

$$\frac{\partial H}{\partial \vec{u}} \cdot \delta \vec{u} = 0 \quad (1.17)$$

$\forall \delta \vec{u}(t)$ variation admissible de la commande

Cas particulier:

Nous détaillerons le cas particulier où Ω est défini par :

$$\Omega = \{ \vec{u}(t) \text{ tel que } \forall t \in [t_0; t_f] \quad g_i(\vec{u}(t)) \leq 0 \quad i=1, n_c \} \quad (1.18)$$

Les variations admissibles seront ici les variations $\delta \vec{u}$ telles que lorsque les contraintes sont saturées:

$$\frac{\partial g_i}{\partial \vec{u}} \cdot \delta \vec{u} = 0 \quad \forall i \text{ tel que } g_i(\vec{u}(t)) = 0 \quad (1.19)$$

Dans ce cas les conditions (1.15)-(1.17) restent valables, à condition de restreindre $\partial H/\partial \vec{u}$ au sous espace tangent à l'espace des contraintes saturées, c'est à dire à l'espace des commande telles que $g_i = 0$.

Interprétation de l'adjoint

En observant (1.12) nous pouvons interpréter la valeur initiale de l'adjoint comme la sensibilité de l'indice de performance J à une variation de l'état initial. Soit :

$$\frac{\partial J}{\partial \vec{x}_0} = \vec{\lambda}_0^T \quad (1.20)$$

L'adjoint peut caractériser la sensibilité du critère optimal vis à vis de variations de l'état à l'instant t .

De façon générale, la sensibilité d'un indice de performance quelconque V vis à vis d'une variation faible de la commande pourra être exprimée par une expression du type (1.12) à l'aide d'un vecteur adjoint associé.

En effet, considérons un indice V :

$$V = \int_{t_0}^{t_f} L(\vec{x}(t), \vec{u}(t), t) dt + \varphi(\vec{x}(t_f), t_f) \quad (1.21)$$

Alors si nous définissons de la même façon que précédemment, l'hamiltonien par (1.5) et l'adjoint associés à cet indice par (1.10) et la conditions terminale:

$$\vec{\lambda}^T(t_f) = \left(\frac{\partial \varphi}{\partial \vec{x}} \right)_{t_f} \quad (1.22)$$

la variation δV sera donnée par :

$$\delta V = \int_{t_0}^{t_f} \left(\frac{\partial H}{\partial \vec{u}} \cdot \delta \vec{u} \right) dt \quad (1.23)$$

Ce résultat essentiel sera largement utilisé dans les méthodes numériques d'optimisation du type gradient que nous verrons ultérieurement.

Définition : Par la suite, nous appellerons trajectoire extrémale toute trajectoire vérifiant les équations (1.14) à (1.17).

1.2.2. Conditions du second ordre

Considérons des variations $\delta \vec{u}$ et $\delta \vec{x}$ autour d'une trajectoire extrémale. Ces variations vérifient le système linéaire :

$$\delta \dot{\vec{x}} = \frac{\partial \vec{f}}{\partial \vec{x}} \delta \vec{x} + \frac{\partial \vec{f}}{\partial \vec{u}} \delta \vec{u} \quad (1.24)$$

$$\delta \vec{x}(t_0) = 0 \quad (1.25)$$

$$\left(\begin{array}{c} \frac{\partial \Psi}{\partial \vec{x}} \delta \vec{x} \\ \frac{\partial \Psi}{\partial \vec{u}} \delta \vec{u} \end{array} \right)_{t_f} = 0 \quad (1.26)$$

Dans ce cas la variation au second ordre de \bar{J} et donc de J sera :

$$\delta^2 \bar{J} = \frac{1}{2} \int_{t_0}^{t_f} (\delta \vec{x}^T \ \delta \vec{u}^T) \cdot \begin{pmatrix} \mathbf{Q} & \mathbf{S} \\ \mathbf{S}^T & \mathbf{R} \end{pmatrix} \cdot \begin{pmatrix} \delta \vec{x} \\ \delta \vec{u} \end{pmatrix} dt + \frac{1}{2} \left(\delta \vec{x}^T \cdot \frac{\partial^2 \Phi}{\partial \vec{x}^2} \cdot \delta \vec{x} \right)_{t_f} \quad (1.27)$$

où
$$\mathbf{Q} = \frac{\partial^2 H}{\partial \vec{x}^2} \quad (1.28)$$

$$\mathbf{S} = \frac{\partial^2 H}{\partial \vec{x} \partial \vec{u}} \quad \mathbf{S}^T = \frac{\partial^2 H}{\partial \vec{u} \partial \vec{x}} \quad (1.29)$$

$$\mathbf{R} = \frac{\partial^2 H}{\partial \vec{u}^2} \quad (1.30)$$

On montre alors le résultat suivant :

Soit le problème linéaire quadratique auxiliaire associé défini par les équations (1.24),(1.25),(1.26) et (1.27) et qui consiste à minimiser $\delta^2 \bar{J}$.

Alors la trajectoire extrémale considérée est un minimum local pour des variations faibles de la commande si et seulement si ce problème admet une solution.

Cette condition nécessaire et suffisante d'optimalité locale se traduit par les trois conditions suivantes :

i) condition de normalité donnée par (H1)

ii) condition de Legendre-Clebsch
$$\frac{\partial^2 H}{\partial \vec{u}^2} > 0 \quad (1.31)$$

iii) condition de non existence de points singuliers
$$\left(\frac{\partial^2 \Phi}{\partial \vec{x}^2} \right)_{\text{f}} \geq 0 \quad (1.32)$$

$$\left(\frac{\partial^2 H}{\partial \vec{x}^2} \right)_{\text{f}} \geq 0 \quad (1.33)$$

où le signe $>$ signifie ici que la matrice est définie positive et \geq qu'elle est semi définie positive.

1.3.Principe de PONTRYAGIN : une condition nécessaire d'optimalité

Dans les années soixante, Pontryagin démontre le principe suivant :

une condition nécessaire pour qu'une commande \vec{u}^* admissible soit optimale est qu'il existe une forme linéaire $\vec{\lambda}(t)$ et un vecteur de multiplicateurs \vec{v} tels que le long de la trajectoire \vec{x}^* engendrée par la commande \vec{u}^* , on ait

$$\forall t, v(t) \in \Omega$$

$$H(\vec{x}^*(t), \vec{u}^*(t), \vec{\lambda}(t), t) < H(\vec{x}(t), \vec{v}(t), \vec{\lambda}(t), t)$$

$$\vec{\lambda}^T = - \frac{\partial H^*}{\partial \vec{x}} \quad (1.34)$$

$$\vec{\lambda}^T(t_f) = \frac{\partial \Phi}{\partial \vec{x}}(\vec{x}^*(t_f), t_f, \vec{v}) \quad (1.35)$$

$$H^*(\vec{x}(t), \vec{\lambda}(t), t) \equiv \min_u H(\vec{x}(t), \vec{u}(t), \vec{\lambda}(t), t) \quad (1.36)$$

En particulier, si le domaine des commandes admissibles est ouvert et si H est deux fois différentiable en x et u, on retrouve les conditions pour qu'une commande soit optimale :

$$\begin{aligned} \frac{\partial H}{\partial \vec{u}} &= 0 \\ \frac{\partial^2 H}{\partial \vec{u}^2} &> 0 \end{aligned}$$

Les équations (1.35), (1.36), (1.37) associées aux équations de la dynamique (1.1) et aux conditions initiales et finales (1.2), (1.3) forment un système d'équations différentielles non linéaires avec conditions aux deux bouts. La résolution de ce problème que nous noterons (P2) permet d'obtenir la commande optimale en fonction de l'état et de l'adjoint. En général, il sera nécessaire de recourir à des algorithmes itératifs pour résoudre ce problème.

Dans le cas particulier du problème linéaire quadratique, la solution de (P2) est analytique et on montre qu'il s'agit de l'optimum.

Définition : nous appellerons fonction hamiltonien optimum, la fonction :

$$H^*(\vec{x}(t), \vec{\lambda}(t), t) = \min_u H(\vec{x}(t), \vec{u}(t), \vec{\lambda}(t), t)$$

1.4. Equation de HAMILTON-BELLMAN-JACOBI: une condition suffisante d'optimalité

Dans les années 50 Bellman, s'appuyant sur le principe d'optimalité, développe une méthode de résolution des problèmes de commande optimale connue sous le nom de programmation dynamique. Il énonce le principe d'optimalité suivant :

quels que soient l'état initial et les commandes initiales,
les commandes restantes doivent constituer une politique
optimale par rapport à l'état qui résulte des commandes initiales.

Appliquant ce principe à l'étude de la commande optimale, il en déduit une condition suffisante d'optimalité. Le raisonnement est le suivant.

Pour un état initial \vec{x} et un instant initial t donnés, il introduit la fonction de retour optimale :

$$V(\vec{x}, t) = \left(\varphi(\vec{x}^*, t) \right)_{tf} + \int_t^{t_f} L(\vec{x}^*, \vec{u}^*, \tau) . d\tau \quad (1.37)$$

où \vec{x}^* et \vec{u}^* sont respectivement la trajectoire et la commande optimale pour ces conditions initiales. Nous aurons de manière immédiate :

$$V(\vec{x}^*(t_f), t_f) = \left(\varphi(\vec{x}^*, t) \right)_{tf}$$

pour $\vec{x}^*(t_f) \in S(t_f)$

Supposons que V est continûment différentiable et considérons la commande suivante :

$$\forall \tau \in [t, t+\Delta t] \quad \vec{u}(\tau) = \vec{u}^o(\tau) \text{ quelconque}$$

$$\forall \tau \in [t+\Delta t, t_f] \quad \vec{u}(\tau) = \vec{u}^*(\tau)$$

où $\vec{u}^*(t)$ est la commande optimale pour les conditions initiales obtenues à $t+\Delta t$ après avoir appliqué la commande $u^o(t)$. Alors, d'après le principe d'optimalité, nous aurons:

$$V(\vec{x}, t) = \min_u \left\{ L(\vec{x}, \vec{u}, t) . \Delta t + V(\vec{x} + \Delta \vec{x}, t + \Delta t) \right\}$$

ce qui donne au premier ordre :

$$V(\vec{x}, t) \approx \min_u \left\{ L(\vec{x}, \vec{u}, t) . \Delta t + V(\vec{x}, t) + \frac{\partial V}{\partial \vec{x}}(\vec{x}, t) . \Delta \vec{x} + \frac{\partial V}{\partial t}(\vec{x}, t) . \Delta t \right\}$$

En divisant par Δt , en faisant tendre Δt vers 0 et en séparant les termes indépendants de la commande, nous obtenons l'équation aux dérivées partielles de HAMILTON-BELLMAN-JACOBI :

$$-\frac{\partial V}{\partial t}(\vec{x}, t) = \min_u \left\{ L(\vec{x}, \vec{u}, t) + \frac{\partial V}{\partial \vec{x}}(\vec{x}, t) . f(\vec{x}, \vec{u}, t) \right\} \quad (1.38)$$

avec la condition

$$V(\vec{x}(t), t) = \varphi(\vec{x}, t) \quad \forall \vec{x}(t) \in S(t) \quad (1.39)$$

On démontre la condition suffisante d'optimalité globale suivante:

S'il existe une fonction $V(\vec{x}(t), t)$ continuellement différentiable solution de l'équation (1.38) telle que la commande associée transfère le système de l'état (\vec{x}_0, t_0) à l'état $(\vec{x}_f, t_f) \in S(t_f)$ alors cette commande $\vec{u}^*(t)$ est optimale parmi toutes les commandes admissibles.
La valeur optimale de J sera $V(\vec{x}_0, t_0)$

Démonstration :

Nous remarquons que le long de la trajectoire $\dot{\vec{x}} = f$ donc

$$\frac{\partial V}{\partial t} + \frac{\partial V}{\partial \vec{x}} \cdot f = \frac{dV}{dt} \quad (\text{dérivée évaluée le long de la trajectoire})$$

Si V est solution de l'équation (1.38), nous aurons pour toute commande admissible qui transfère l'état de (x_0, t_0) à $(x_f, t_f) \in S(t_f)$

$$\frac{dV}{dt} + L \geq 0$$

En intégrant de t_0 à t_f et compte tenu de la condition finale, nous obtenons

$$J - V(\vec{x}_0, t_0) \geq 0$$

quelque soit la commande admissible qui transfère l'état de (\vec{x}_0, t_0) à l'état $(\vec{x}_f, t_f) \in S(t_f)$. La commande associée à cette solution est donc bien la commande optimale.

Remarque :

D'après la remarque précédente sur la signification de l'adjoint initial, nous avons :

$$\frac{\partial V}{\partial \vec{x}} \equiv \vec{\lambda}^T \quad (1.40)$$

et l'équation (1.38) n'est autre que la minimisation de l'hamiltonien H. Ce qui veut dire que :

$$H^* \equiv - \frac{\partial V}{\partial t}(\vec{x}, t) \quad (1.41)$$

L'équation (1.38), lorsqu'elle admet une solution, permet d'obtenir une commande par retour d'état. Cependant, outre le fait que la solution n'existe pas toujours, la résolution de cette équation aux dérivées partielles non linéaire se révèle très délicate sauf pour certains cas particuliers (comme les problèmes linéaires quadratiques).

Les méthodes basées sur cette théorie ne viseront pas, en général, à résoudre l'équation mais l'utiliseront comme "outil" de calcul. Nous citerons par exemple, les algorithmes proposés par Mitter (1966) ou par Jacobson & Mayne (1970).

1.5. Traitement des contraintes

Jusqu'à présent, nous n'avons pas considéré les contraintes éventuelles sur l'état et la commande en dehors des contraintes finales. Or beaucoup de systèmes sont soumis à différents types de contraintes courantes : limitation des commandes, contraintes de structures, domaine d'évolution restreint etc... L'étude des contraintes n'étant pas l'objet de cette thèse, la présentation qui suit ne sera pas complète. Nous ne donnerons que les principales contraintes rencontrées dans la littérature et les conditions d'optimalité rattachées à ces contraintes. Elles seront classées en trois groupes selon qu'elles s'appliquent sur l'état, la commande ou simultanément sur l'état et la commande. Nous réserverons un paragraphe particulier pour les contraintes de type intégrale.

Nous appellerons contrainte courante une contrainte qui s'appliquera sur tout l'intervalle de temps $[t_0; t_f]$.

Nous appellerons contrainte égalité, une contrainte du type :

$$C = 0$$

(C peut dépendre du temps et de la commande: $C(\vec{u}, t)$, du temps et de l'état: $C(\vec{x}, t)$ ou simultanément de ces trois variables: $C(\vec{u}, \vec{x}, t)$)

Nous appellerons contraintes inégalité, des contraintes du type :

$$C \leq 0$$

Pour ces contraintes inégalité, nous dirons qu'elles sont saturées ou actives lorsqu'il y aura égalité, c'est à dire lorsque $C = 0$. Les intervalles de temps sur lesquels la contrainte sera saturée seront dit saturés. De même, les portions de trajectoire correspondant à ces intervalles de temps seront des arcs saturés.

1.5.1. Contraintes sur la commande

Nous avons déjà évoqué ces contraintes précédemment. Elles sont en fait implicitement définies par le domaine des commandes admissibles. Nous examinerons particulièrement le cas où le domaine admissible sera défini par :

$$\Omega = \{ \vec{u} \text{ t.q. } g_i(\vec{u}, \vec{x}) \leq 0 \quad i = 1 \dots n_c \}$$

avec n_c contraintes $n_c \leq n_u$

et $\frac{\partial g_i}{\partial u_i} \neq 0$

$$\frac{\partial g_i}{\partial u_j} = 0 \quad j \neq i$$

Pour $\vec{u}(t)$ sur la frontière de Ω , la restriction des conditions d'optimalité à l'espace tangent aux contraintes saturées peut être traitée de la façon suivante.

Soit l'hamiltonien augmenté

$$\bar{H} = H + \sum_{j \in J} \mu_j \cdot g_j(\vec{u}, \vec{x}) \quad (1.42)$$

où J est l'ensemble des indices des contraintes saturées

Alors les conditions d'optimalité seront

$$\frac{\partial \bar{H}}{\partial \vec{u}}(\vec{u}, \vec{x}, \vec{\mu}, \vec{\lambda}) = 0 \quad (1.43)$$

$$\frac{\partial^2 \bar{H}}{\partial \vec{u}^2}(\vec{u}, \vec{x}, \vec{\mu}, \vec{\lambda}) > 0 \quad (1.44)$$

$$\vec{\mu} = (\mu_1, \dots, \mu_{n_j})^T \text{ tels que } g_j(\vec{u}, \vec{x}) = 0 \quad j \in J \quad (1.45)$$

et l'adjoint sera défini par

$$\vec{\lambda}^T = - \frac{\partial \bar{H}}{\partial \vec{x}} \quad (1.46)$$

soit
$$\vec{\lambda}^T = - \frac{\partial H}{\partial \vec{x}} - \sum_{j \in J} \mu_j \cdot \frac{\partial g_j}{\partial \vec{x}}(\vec{u}, \vec{x}) \quad (1.47)$$

En fait, cela revient à appliquer le principe du minimum avec l'hamiltonien augmenté et avec μ_i tel que

$$\mu_i = 0 \text{ si } g_i < 0$$

$$\mu_i \geq 0 \text{ lorsque la contrainte } g_i \text{ est active.}$$

Remarque :

Dans le cas particulier où les contraintes sur les commandes sont des bornes du type $u_{im}(\vec{x}) \leq u_i \leq u_{iM}(\vec{x})$

nous définissons les fonction g_i par $g_i(\vec{u}, \vec{x}) = (u_{iM}(\vec{x}) - u_i) \cdot (u_i - u_{im}(\vec{x}))$

et le multiplicateur associé à g_i lorsque la contrainte est saturée sera donné par:

$$\mu_i = \frac{\frac{\partial H}{\partial u_i}(\vec{x}, u_{opt}, \lambda, t)}{\varepsilon \left(u_{im}(\vec{x}) - u_{iM}(\vec{x}) \right)} \quad (1.48)$$

où $\varepsilon = 1$ si $u_i = u_{iM}$
 où $\varepsilon = -1$ si $u_i = u_{im}$

1.5.2. Contraintes sur l'état

Dans la pratique ce sont les contraintes les plus difficiles à traiter. Les contraintes égalités réduisent la dimension du système et sont en fait rencontrées lors de saturation de contraintes inégalités. Nous présentons ici les conditions liées à ces contraintes -inégalités données par

$$C(\vec{x}(t)) \leq 0 \quad (1.49)$$

où C est une fonction différentiable par rapport à ses arguments.

Définitions :

q sera l'ordre de la contrainte si

$$\forall i < q \quad \frac{\partial}{\partial \vec{u}} \left(\frac{d^i}{dt^i} C(\vec{x}(t)) \right) = 0$$

$$\frac{\partial}{\partial \vec{u}} \left(\frac{d^q}{dt^q} C(\vec{x}(t)) \right) \neq 0$$

Le vecteur des contraintes de saturation sera :

$$\vec{N} = \left(\frac{d^0 C}{dt^0}, \frac{d^1 C}{dt^1}, \dots, \frac{d^{q-1} C}{dt^{q-1}} \right)^T$$

Et l'hamiltonien augmenté :

$$\bar{H} = H + \mu \cdot \left(\frac{d^q C}{dt^q} \right)$$

Nous aurons alors les conditions d'optimalité suivantes :

1) Sur un intervalle saturé $[t_1; t_2]$

$$\forall t \in [t_1; t_2] \quad \vec{N}(t) = 0$$

$$\exists \vec{\sigma} \text{ tel que } \sigma_i \geq 0 \quad \forall i=0 \dots q-1 \text{ et}$$

$$\vec{\lambda}^T(t_1^+) = \vec{\lambda}^T(t_1^-) - \vec{\sigma}^T \cdot \frac{\partial C}{\partial \vec{x}}(t_1) \quad (1.50)$$

$$\vec{\lambda}^T(t_2^+) = \vec{\lambda}^T(t_2^-) \quad (1.51)$$

$\mu(t)$ est de classe C^q sur $[t_1; t_2]$ tel que

$$\frac{d^i \mu}{dt^i}(t) \geq 0 \quad (1.52)$$

$$\frac{d^i \mu}{dt^i}(t_1^+) = \sigma_{q-1-i} \text{ sauf pour } i = q-1 \text{ où } \frac{d^i \mu}{dt^i}(t_1^+) \geq \sigma_0 \quad (1.53)$$

$$\frac{d^i \mu}{dt^i}(t_2^-) = 0 \text{ sauf pour } i = q-1 \text{ où } \frac{d^i \mu}{dt^i}(t_2^-) \geq 0 \quad (1.54)$$

$$H \text{ est continue en } t_1 \text{ et } t_2 \quad (1.55)$$

2) Pour un point isolé t_i où $C(\vec{x}(t_i)) = 0$ et $C(\vec{x}(t)) \leq 0$ sur un voisinage de t_i

H est continue et

$$\exists \sigma_0 \geq 0 \quad \text{tel que}$$

$$\vec{\lambda}^T(t_i^+) = \vec{\lambda}^T(t_i^-) - \sigma_0 \cdot \frac{\partial C}{\partial \vec{x}}(t_i) \quad (1.56)$$

1.5.3. Contraintes sur l'état et la commande

Ces contraintes sont en fait des contraintes sur l'état d'ordre zéro. L'application des conditions précédentes montre qu'à l'entrée et à la sortie d'un intervalle où la contrainte est saturée, l'adjoint, l'hamiltonien et le multiplicateur sont continus avec $\mu \geq 0$.

Dans le cas particulier où la commande est scalaire, la restriction des variations du premier ordre à l'espace tangent aux contraintes saturées se traduira par des variations admissibles telles que :

$$\delta \vec{u} = - \left(\frac{\partial C}{\partial \vec{u}} \right)^{-1} \cdot \left(\frac{\partial C}{\partial \vec{x}} \cdot \delta \vec{x} \right) \quad (1.59)$$

1.5.4. Contraintes intégrales

Définition : nous appellerons contrainte intégrale une contrainte du type

$$\varphi^1(\vec{x}(t_f), t_f) + \int_{t_0}^{t_f} L^1(\vec{x}(t), \vec{u}(t), t) dt \leq 0 \quad (1.60)$$

En fait cette contrainte n'est difficile à traiter que lorsqu'il y a égalité. On peut dans ce cas la traiter simplement en augmentant la dimension du système. En effet, si nous posons

$$\begin{aligned} \overset{\circ}{x}_{n_x+1}(t) &= L^1(\vec{x}(t), \vec{u}(t), t) \\ \overset{\circ}{x}_{n_x+1}(t_0) &= 0 \end{aligned}$$

la contrainte s'écrit :

$$\varphi^1(\vec{x}(t_f), t_f) + \overset{\circ}{x}_{n_x+1}(t_f) = 0$$

Nous revenons donc à la formulation classique du problème où les contraintes sont uniquement finales. Il est facile de montrer que l'adjoint associé à $\overset{\circ}{x}_{n_x+1}$ est une constante et nous aurons :

$$H = L + \vec{\lambda}^T \vec{f} + \mu \cdot L^1 \quad (1.61)$$

avec

$$\vec{\lambda}^T(t_f) = \frac{\partial \varphi}{\partial \vec{x}} + \vec{v}^T \cdot \frac{\partial \psi}{\partial \vec{x}} + \mu \frac{\partial(\varphi^1 + \overset{\circ}{x}_{n_x+1})}{\partial \vec{x}} \quad (1.62)$$

Remarque :

D'une manière générale, les contraintes courantes portant sur l'état sont difficiles à traiter. Elles pourront être ramenées plus ou moins artificiellement à des contraintes finales soit par une méthode du type fonction de pénalité soit en les ramenant à des contraintes intégrales. En effet, on peut exprimer toutes les contraintes d'inégalité courante du type $W \leq 0$ sous la forme d'une contrainte intégrale en utilisant l'artifice

de calcul suivant:

$$V = \int_{t_0}^{t_f} M dt = 0$$

où

$$M = 0 \quad \text{si } W \leq 0$$
$$M = W^2 \quad \text{si } W > 0$$

Sous l'hypothèse de continuité de W , cette nouvelle contrainte finale égalité sera équivalente à la contrainte courante.

2. METHODES NUMERIQUES D'OPTIMISATION

Dans le cas des systèmes non linéaires, les équations à résoudre dans un problème de commande optimale sont en général très complexes. Dans la plupart des cas, il est impossible de trouver une solution analytique. De ce fait, les nombreuses méthodes publiées à ce jour sont soit des méthodes numériques, soit des méthodes approchées. Cette dernière approche fait d'ailleurs l'objet de recherches plus récentes.

Le but de cette partie est de recenser les méthodes de résolutions développées à ce jour. Sans prétendre en donner une liste exhaustive, nous présenterons les principales méthodes d'optimisation numériques trouvées dans la littérature. Nous les avons classées selon quatre groupes. Le premier rassemble les méthodes directes basées sur le calcul de variations. Parmi les méthodes indirectes basées sur la résolution du problème de Pontryagin, nous avons distingué les méthodes du type paramétrique et les méthodes du type fonctionnel. Le troisième groupe concerne les méthodes de calcul par retour d'état. Nous avons cité dans le dernier groupe, les méthodes approchées par paramétrisation.

Les méthodes directes modifient la commande de manière itérative, afin de diminuer le critère J , tout en respectant les contraintes. Ces méthodes utilisent les résultats du calcul de variation et sont du type gradient. Ces méthodes assurent à chaque itération une amélioration de l'indice de performance mais ne cherchent pas à résoudre explicitement les équations d'optimalité.

Pour les méthodes indirectes la commande est éliminée des équations d'Euler-Lagrange en résolvant soit $\partial H / \partial \bar{u}^E = 0$, soit $\bar{u}^E = \text{Argmin } H$. Il s'agit alors de résoudre par un algorithme itératif le "problème avec conditions aux deux bout" constitué par (P2). Ces méthodes ont en général une convergence rapide, en particulier ce sont souvent des méthodes à convergence quadratique. Parmi ces méthodes, on trouve les méthodes de tir et de voisinage dont les itérations sont basées sur la correction des conditions initiales ou finales et les méthodes de quasilinearisation qui modifient à chaque itération directement les fonctions d'état et les fonctions adjointes.

Parmi les méthodes de calcul d'une commande par retour d'état, l'une des premières est la programmation dynamique différentielle basée sur l'équation de HAMILTON-BELLMAN-JACOBI. Ce type d'approche a l'avantage de donner une commande en boucle fermée mais reste malheureusement d'un emploi assez limité. On

trouve cependant pour certains cas particuliers, des méthodes qui donnent une solution analytique ou une solution approchée simple à calculer. Ces méthodes utilisent soit des développements particuliers, soit le formalisme des crochets de Lie.

Une dernière approche est celle des méthodes paramétriques. Ce sont des méthodes approchées. Celles ci consistent à exprimer la commande ou les équations du problème par des expressions qui dépendent d'un certain nombre de paramètres. Ces paramètres seront calculés de façon à optimiser le critère tout respectant un certain nombre de contraintes supplémentaires imposées par les approximations choisies.

2.1.Méthodes du gradient

Les premières méthodes (Kelley 1960) ont été développées pour la résolution des problèmes sans contraintes. Ces méthodes basées sur le principe du gradient simple ont été généralisées pour traiter des problèmes soumis à diverses contraintes. Bryson & Denham (1965) proposent une méthode de gradient projeté qui traite les contraintes finales. Lasdon et al. (1967) et Hestenes (1970) utilisent la méthode des gradients conjugués. Miele (1970) traite les contraintes courantes égalités sur l'état et la commande par une méthode de "recalage séquentiel", Aumasson (1980) traite des problèmes très généraux par une méthode de gradient projeté. Ces méthodes du premier ordre convergent très lentement au voisinage de l'optimum. Pour accélérer la convergence, certains auteurs utilisent les variations du second ordre (Kelley et al. 1960, Mitter 1966), d'autres modifient les méthodes du gradient en utilisant la minimisation de l'hamiltonien (Gottlieb 1967, Blank & Shinar 1976). Cette revue des méthodes du gradient n'est bien sûr pas exhaustive, nous ne donnons ici que quelques une des méthodes représentatives.

2.1.1. Méthodes du premier ordre

Il s'agit de calculer, à partir d'une commande nominale, des variations de la commande qui permettent, d'une itération à l'autre, de réduire l'indice de performance tout en respectant les contraintes éventuelles. Ces variations seront d'une manière générale de la forme

$$\vec{u}(t)^{N+1} = \vec{u}(t)^N - \alpha \left(\frac{\partial \vec{H}^N}{\partial \vec{u}} \right)^T \quad (1.63)$$

où $(.)^N$ indique que les fonctions sont prises à l'itération N
et où α est un réel positif

La vitesse de convergence de l'algorithme dépendra du coefficient α . En effet la variation au premier ordre de l'indice de performance sera alors

$$\delta J = - \alpha \int_{t_0}^{t_f} \left(\frac{\partial H}{\partial \vec{u}} \cdot \frac{\partial H}{\partial \vec{u}}^T \right) dt \quad (2.2)$$

et J sera alors d'autant plus diminué que ce coefficient sera grand. Cependant l'algorithme étant basé sur les calculs de variation du premier ordre la variation ne devra pas être trop importante. Il faut donc trouver un bon compromis entre ces deux impératifs.

Définition : nous appellerons pas de l'algorithme la norme de la variation de commande d'une itération à l'autre , c'est à dire

$$N \left(\vec{u}(t)^{N+1} - \vec{u}(t)^N \right)$$

$$\text{avec } N(.) = \left(\int_{t_0}^{t_f} \| \cdot \|_G^2 dt \right)^{1/2}$$

$$\| \cdot \|_G^2 = (.)^T \cdot G \cdot (.) \quad G = \text{diag} (1/G_i)$$

où G_i est un scalaire strictement positif

Nous présentons ici, deux méthodes parmi les nombreuses méthodes existantes.

1) Méthode de gradient séquentiel recalé :

Miele (1970) propose un algorithme de gradient séquentiel recalé qui permet de traiter les contraintes courantes d'égalité du type $C(\vec{x}, \vec{u}) = 0$. Shinar (1978) élargit son domaine d'application aux contraintes d'inégalité du type $C(\vec{x}, \vec{u}) \leq 0$

Le principe de cette méthode est le suivant : une itération se fera en deux phases. On suppose qu'à l'itération N, les contraintes sont respectées.

La première phase consiste à calculer une variation de commande telle que les contraintes soient respectées au premier ordre.

Sur un intervalle où la contrainte est respectée cette variation sera

$$\delta \vec{u}(t) = - \alpha \cdot \left(\frac{\partial \vec{H}^N}{\partial \vec{u}} \right)^T \quad (2.3)$$

où $\alpha \in]0;1]$

Sur un intervalle où la contrainte est saturée cette variation sera

$$\begin{aligned} \delta \vec{u}(t) &= - \alpha \cdot \left(\frac{\partial \vec{H}^N}{\partial \vec{u}} \right)^T & \text{si } \frac{\partial C}{\partial \vec{u}} \cdot \delta \vec{u} + \frac{\partial C}{\partial \vec{x}} \cdot \delta \vec{x} \leq 0 \\ \delta \vec{u}(t) &= \text{Arg} \left(\frac{\partial C}{\partial \vec{u}} \cdot \delta \vec{u} + \frac{\partial C}{\partial \vec{x}} \cdot \delta \vec{x} \right) & \text{si } \frac{\partial C}{\partial \vec{u}} \cdot \delta \vec{u} + \frac{\partial C}{\partial \vec{x}} \cdot \delta \vec{x} > 0 \end{aligned}$$

Dans une seconde phase, il s'agira de vérifier si les contraintes sont respectées. Si ce n'est pas le cas, la variation de la commande est trop grande pour que l'approximation sur les contraintes linéarisées soient respectées. Il faudra diminuer les erreurs sur la contrainte. Cette phase sera calculée en minimisant

$$\int_{t_0}^{t_f} \|\delta \vec{u}\|_G^2 dt$$

sous la contrainte linéarisée

$$\frac{\partial C}{\partial \vec{u}} \cdot \delta \vec{u} + \frac{\partial C}{\partial \vec{x}} \cdot \delta \vec{x} + \alpha C = 0$$

où $\alpha \in]0;1]$

2) Méthode de gradient projeté généralisé :

Ce programme développé à l'ONERA (Aumasson 1980) traite des problèmes contraints par une méthode de gradient projeté généralisé.

Dans cette méthode, toutes les contraintes courantes sur l'état et sur la commande sont ramenées à des contraintes finales en utilisant la formulation intégrale (cf paragraphe 1.2.1.).

Nous aurons alors n_{cf} contraintes finales notées:

$$V_i = \xi_i(\vec{x}(t_f), t_f) + \int_{t_0}^{t_f} M_i(\vec{x}(t), \vec{u}(t), t) dt = 0 \quad i = 1 \dots n_{cf}$$

Pour le coût et pour chacune de ces contraintes seront définis un hamiltonien et un adjoint associé:

$$H^i = M_i + \vec{\lambda}^{iT} \cdot \vec{f} \quad (2.4)$$

$$\vec{\lambda}^{iT}(t) = - \frac{\partial H^i}{\partial \vec{x}} \quad (2.5)$$

$$\vec{\lambda}^{iT}(t_f) = \frac{\partial \xi_i}{\partial \vec{x}}(t_f) \quad (2.6)$$

où $i = 0 \dots n_c$

et $M_0 = L \quad \xi_0 = \varphi$

Les variations de l'indice de performance et des contraintes dues à une petite variation de la commande seront alors exprimées comme nous l'avons vu en 2.2 par:

$$\delta J = \int_{t_0}^{t_f} \left(\frac{\partial H^0}{\partial \vec{u}} \right) \cdot \delta \vec{u}(t) dt \quad (2.7)$$

$$\delta V_i = \int_{t_0}^{t_f} \left(\frac{\partial H^i}{\partial \vec{u}} \right) \cdot \delta \vec{u}(t) dt \quad (2.8)$$

La variation de commande $\delta \vec{u}(t)$ sera alors solution du problème quadratique

trouver $\delta \vec{u}(t)$ pour que

- les variations des contraintes soient égales à des objectifs δV_i^* qui représentent le "meilleur" rattrapage possible des contraintes compte tenu des hypothèses de calcul au premier ordre
- le pas de l'itération soit minimal, ce qui se traduit par :

$$\int_{t_0}^{t_f} \|\delta \vec{u}(t)\|_G^2 dt \quad \text{minimum.}$$

La solution de ce problème donne:

$$\delta \vec{u}(t) = \mathbf{G}^{-1} \cdot \frac{\partial \vec{H}^T}{\partial \vec{u}} \cdot \Gamma^{-1} \cdot \delta \vec{V}^* \quad (2.9)$$

$$\text{avec } \Gamma = \int_{t_0}^{t_f} \frac{\partial \vec{H}}{\partial \vec{u}} \mathbf{G}^{-1} \frac{\partial \vec{H}^T}{\partial \vec{u}} dt \quad (2.10)$$

où \vec{H} est le vecteur des hamiltoniens associés aux contraintes
 $\vec{H}^T = (H^0 H^1 \dots H^{ncf})$

Toutes ces méthodes ralentissent considérablement au voisinage de l'optimum. En effet, comme $\partial H / \partial \vec{u}$ tend vers 0, le pas de l'algorithme diminuera de même.

Un moyen d'accélérer la convergence consistera à utiliser le minimum de l'hamiltonien. A l'itération N, on calcule la commande \vec{u}^{*N} qui minimise l'hamiltonien pour $\vec{x}^{\rightarrow N}$ et $\vec{\lambda}^{\rightarrow N}$ donnés. La variation de commande sera

$$\delta \vec{u} = \alpha (\vec{u}^{*N} - \vec{u}^{\rightarrow N})$$

où $\vec{u}^{*N} = \text{Argmin } H^N$
 et $0 < \alpha \leq 1$

Une autre manière d'accélérer la convergence de l'algorithme sera d'utiliser des variations du second ordre.

2.1.2. Méthodes du second ordre

L'une des premières méthodes présentée est due à Mitter (1966). Le principe est le suivant.

Soit un multiplicateur \vec{v} , une commande nominale \vec{u} , la trajectoire correspondante et un vecteur adjoint obtenu par intégration inverse de (1.10) à partir des conditions (1.11). La variation de J devient:

$$\delta^2 J = \frac{1}{2} \int_{t_0}^{t_f} \left(\frac{\partial H}{\partial \vec{u}} \cdot \delta \vec{u} + (\delta \vec{x}^T \ \delta \vec{u}^T) \cdot \begin{pmatrix} \mathbf{Q} & \mathbf{S} \\ \mathbf{S}^T & \mathbf{R} \end{pmatrix} \cdot \begin{pmatrix} \delta \vec{x} \\ \delta \vec{u} \end{pmatrix} \right) dt$$

$$+ \frac{1}{2} \left(\delta \vec{x}^T \cdot \frac{\partial^2 \Phi}{\partial \vec{x}^2} \cdot \delta \vec{x} \right)_{t_f} \quad (2.11)$$

$\delta \vec{u}$ sera alors la solution du problème auxiliaire :

minimiser δJ

avec la condition finale sur $\delta \vec{x}$

$$\left(\frac{\partial \vec{\psi}}{\partial \vec{x}} \delta \vec{x} + \vec{\psi} \right)_{t_f} = 0$$

Nous nous trouvons devant la formulation classique du linéaire quadratique. Si la nominale est assez proche de l'optimum, les conditions de normalité, de Legendre-Clebsch et de non existence de point conjugués sont vérifiées et nous pouvons appliquer les résultats classiques de Kalman sur les systèmes linéaires quadratiques.

La solution sera de la forme :

$$\delta \vec{u} = - \mathbf{R}^{-1} \left(\frac{\partial H}{\partial \vec{u}} + \mathbf{S}^T \delta \vec{x} + \frac{\partial \vec{f}^T}{\partial \vec{u}} \delta \vec{\lambda} \right) \quad (2.12)$$

$\delta \vec{x}$ et $\delta \vec{\lambda}$ étant solution du système d'équations différentielles

$$\dot{\delta \vec{x}} = \frac{\partial \vec{f}}{\partial \vec{x}} \delta \vec{x} + \frac{\partial \vec{f}}{\partial \vec{u}} \delta \vec{u} \quad (2.13)$$

$$\dot{\delta \vec{\lambda}} = - \mathbf{Q} \delta \vec{x} + \mathbf{R} \delta \vec{u} + \frac{\partial \vec{f}^T}{\partial \vec{x}} \delta \vec{\lambda} \quad (2.14)$$

avec les conditions initiales et finales

$$\delta \vec{x}(t_0) = 0 \quad (2.15)$$

$$\frac{\partial \vec{\psi}}{\partial \vec{x}_f} \delta \vec{x}(t_f) + \vec{\psi}(\vec{x}(t_f), t_f) = 0 \quad (2.16)$$

$$\delta \vec{\lambda}(t_f) = \frac{\partial^2 \Phi}{\partial \vec{x}_f^2} (\vec{x}(t_f), \vec{v}, t_f) \quad (2.17)$$

Il s'agit, maintenant de résoudre ce système d'équations différentielles avec conditions aux deux bouts.

L'une des méthodes de résolution la plus intéressante du point de vue de la stabilité numérique et de la précision, est celle qui s'inspire de la théorie de Kalman. Sous les hypothèses précédentes, nous pouvons calculer $\vec{\delta u}$ en fonction de l'état et du rattrapage des contraintes $\vec{\delta \psi}$ égal à $-\beta \vec{\psi}$ où β est un coefficient compris entre 0 et 1, réglé pour que les approximations du premier ordre soient valides. Nous aurons

$$\begin{aligned} \vec{\delta u} = & -\mathbf{R}^{-1} \left(\frac{\partial H}{\partial \vec{u}} + \frac{\partial \vec{f}^T}{\partial \vec{u}} \cdot \vec{I} + \frac{\partial \vec{f}^T}{\partial \vec{u}} \mathbf{N} \cdot \vec{\delta v} \right) \\ & - \mathbf{R}^{-1} \left(\mathbf{S}^T + \frac{\partial \vec{f}^T}{\partial \vec{u}} \cdot \mathbf{K} \right) \vec{\delta x} \end{aligned} \quad (2.18)$$

$$\vec{\delta v} = \mathbf{P}_o^{-1} \left(\vec{\delta \psi} - \vec{m}_o \right) \quad (2.19)$$

où \vec{I} , \vec{m} , \mathbf{K} , \mathbf{N} , \mathbf{P} et \mathbf{R} sont les solutions d'équations différentielles intégrées en sens inverse à partir de conditions finales connues.

Cette méthode est très intéressante du point de vue de la forme de la commande. Nous avons une variation de commande qui ne dépend plus que des rattrapages sur la contraintes et de la variation d'état ce qui permet des corrections de \vec{u} en boucle fermée.

Cependant il faudra disposer pour initialiser l'algorithme d'une solution déjà proche de l'optimum (afin de respecter les hypothèses de kalman). De plus les calculs au second ordre nécessitent un volume de calcul important.

Gottlieb (1967) propose une approche plus simple que celle du second ordre qui associe les calculs de variations avec la minimisation de l'hamiltonien. Cette méthode permet d'avoir une bonne rapidité de convergence avec une sensibilité moindre à l'initialisation. Récemment Kelly (1990), reprend cette approche pour un algorithme de guidage en utilisant des dérivées partielles analytiques.

2.2.Méthodes des extrémiaux voisins et méthodes de tir

Les méthodes des extrémiaux voisins (Pesch 1989) et les méthodes de tir (Blank & Shinar, 1978, Grimm & Berger 1985) calculent la solution à partir d'une trajectoire optimale dont les conditions finales et initiales sont voisines des conditions initiales et finales désirées.

Les méthodes des extrémiaux voisins cherchent la variation de commande qui correspond à la variation désirée des conditions initiales et finales. Les méthodes de tir consisteront à modifier l'adjoint initial et les multiplicateurs de lagrange afin de satisfaire les conditions finales que l'on veut atteindre.

2.2.1. Méthode des extrémiaux voisins :

A partir d'une trajectoire nominale optimale correspondant à des conditions initiales et finales:

$$\begin{aligned}\vec{x}(t_0) &= \vec{x}_0 \\ \vec{\Psi}(\vec{x}_f, t_f) &= 0\end{aligned}$$

et vérifiant les contraintes courantes $C(\vec{x}, \vec{u}) \geq 0$, Pesch considère des perturbations $\delta\vec{x}_0$ et $d\vec{\Psi}$. La trajectoire optimale voisine correspondant aux conditions initiales et finales perturbées sera alors calculée en résolvant le système obtenu par linéarisation des conditions optimales autour de la trajectoire nominale:

$$\delta\vec{x} = \frac{\partial \vec{f}}{\partial \vec{x}} \cdot \delta\vec{x} + \frac{\partial \vec{f}}{\partial \vec{u}} \cdot \delta\vec{u} \quad (2.19)$$

$$\delta\vec{\lambda} = -\frac{\partial^2 H}{\partial \vec{x}^2} \cdot \delta\vec{x} - \frac{\partial^2 H}{\partial \vec{x} \partial \vec{u}} \cdot \delta\vec{u} - \frac{\partial \vec{f}^T}{\partial \vec{x}} \cdot \delta\vec{\lambda} - \frac{\partial C^T}{\partial \vec{x}} \cdot \delta\mu \quad (2.20)$$

$$0 = \frac{\partial^2 H}{\partial \vec{u} \partial \vec{x}} \cdot \delta\vec{x} + \frac{\partial^2 H}{\partial \vec{u}^2} \cdot \delta\vec{u} + \frac{\partial \vec{f}^T}{\partial \vec{u}} \cdot \delta\vec{\lambda} + \frac{\partial C^T}{\partial \vec{u}} \cdot \delta\mu \quad (2.21)$$

$$\delta\vec{x}(t_0) = \delta\vec{x}_0 \quad (2.22)$$

$$d\vec{\psi} = \frac{\partial \vec{\psi}}{\partial \vec{x}} \cdot d\vec{x} + \frac{\partial \vec{\psi}}{\partial t_f} \cdot dt_f \quad (2.23)$$

$$d\vec{\lambda}(t_f) = \frac{\partial^2 \vec{\Phi}}{\partial \vec{x}^2} \cdot d\vec{x} + \frac{\partial \vec{\psi}^T}{\partial \vec{x}} \cdot d\vec{v} + \frac{\partial^2 \vec{\Phi}}{\partial \vec{x} \partial t} \cdot dt_f \quad (2.24)$$

$$0 = \left(\frac{\partial^2 \vec{\Phi}}{\partial \vec{x} \partial t} - \dot{\vec{\lambda}}^T \right) \cdot d\vec{x} + \dot{\vec{x}} \cdot d\vec{\lambda} + \frac{\partial \vec{\psi}^T}{\partial t} \cdot d\vec{v} + \frac{\partial^2 \vec{\Phi}}{\partial t^2} \cdot dt_f \quad (2.25)$$

En considérant l'instant initial comme instant courant, la résolution de ce système différentiel linéaire par par une méthode paramétrique donne une variation de commande de la forme:

$$\delta \vec{u}(t) = \Lambda_1(t) \delta \vec{x}(t) + \Lambda_2(t) d\vec{\psi}(t) \quad (2.26)$$

et on obtient une trajectoire optimale voisine de la nominale et correspondant à des perturbations $\delta \vec{x}(t)$ et $\delta \vec{\psi}(t)$ données, si celles ci restent petites.

2.2.2.Méthode de tir direct :

Les méthodes de tir supposent que l'on dispose d'une valeur nominale de l'adjoint initial et du multiplicateur de lagrange. Il s'agira d'intégrer le système (1.1),(1.34) à partir de valeurs initiales \vec{x}_0 et $\vec{\lambda}_0$, la commande étant obtenue par minimisation de l'hamiltonien. Les conditions finales n'étant pas vérifiées à priori, la procédure utilisée consistera à recalculer de façon itérative l'adjoint initial $\vec{\lambda}_0$ et le vecteur \vec{v} jusqu'à ce que les conditions finales soient respectées.

Blank et Shinar proposent en 1978 une méthode appelée méthode de tir direct qui consiste à résoudre l'équation implicite

$$E(\lambda_0, v) = 0 \quad \text{où } E = \left(\|\vec{\psi}\|^2 + \|\vec{\lambda}^T - \frac{\partial \vec{\Phi}}{\partial \vec{x}}\|^2 \right)_{tf} \quad (2.27)$$

en minimisant E par la méthode des directions conjuguées.

Le point sensible de cette méthode viendra du calcul de la matrice de sensibilité des conditions terminales vis à vis de variations de $\vec{\lambda}_0$ et de \vec{v} ; c'est à dire la matrice que nous noterons d'un point de vue formel

$$\frac{\partial \left(\vec{\psi}, \vec{\lambda}_f - \frac{\partial \vec{\Phi}}{\partial \vec{x}_f} \right)}{\partial (\vec{\lambda}_0, \vec{v})} \quad (2.28)$$

Le calcul de cette matrice sera en général très sensible aux erreurs numériques. De plus l'équation adjointe étant intégrée en sens direct, il sera nécessaire d'avoir une valeur initiale de l'adjoint relativement proche de la valeur optimale. Si ce n'est pas le cas, l'intégration peut donner à l'instant final des valeurs très importantes qui risquent même de dépasser les capacités du calculateur.

2.2.3.Méthode de tir multiple :

Afin d'éviter la divergence due à l'intégration en sens direct du système adjoint, Grimm et Berger (1985) recalculent l'adjoint et l'état à intervalles réguliers. Le problème de tir direct est remplacé par une série de "tirs" définis sur des sous intervalles de temps $[t_i; t_{i+1}]$ et reliés entre eux par des relations de continuité.

Il s'agira de calculer les paramètres

$$\vec{\lambda}_0, \vec{x}_1, \vec{\lambda}_1, \vec{x}_2, \vec{\lambda}_2, \dots, \vec{x}_f, \vec{\lambda}_f$$

où $\vec{x}_i, \vec{\lambda}_i$ sont les valeurs estimées à l'instant t_i

tels que

$$\vec{\psi}(\vec{x}_f, t_f) = 0 \quad (2.29)$$

$$\vec{\lambda}_f - \left(\frac{\partial \vec{\Phi}}{\partial \vec{x}} \right)_{t_f} = 0 \quad (2.30)$$

$$C(\vec{\lambda}_0, \vec{x}_1, \vec{\lambda}_1, \vec{x}_2, \vec{\lambda}_2, \dots, \vec{x}_f, \vec{\lambda}_f) = 0 \quad (2.31)$$

où C représente toutes les conditions de continuité associées à chaque $\vec{x}_i, \vec{\lambda}_i$ et t_i .

Ces conditions seront du type:

$$\begin{pmatrix} \vec{x}_{i+1} \\ \vec{\lambda}_{i+1} \end{pmatrix} - \begin{pmatrix} \vec{x}_i \\ \vec{\lambda}_i \end{pmatrix} - \int_{t_i}^{t_{i+1}} \begin{pmatrix} \vec{f}^* \\ \partial H^* / \partial \vec{x} \end{pmatrix} dt = 0 \quad (2.32)$$

H^* étant l'hamiltonien optimum

et $\vec{f}^* = \vec{f}(\vec{x}, \vec{u}^*, t)$

Ces méthodes de tir se révèlent plus facile à mettre en œuvre que les méthodes de gradient et le volume de calcul requis est relativement peu important. Cependant l'estimation initiale de $\vec{\lambda}_0$ et de \vec{v} sera souvent difficile. Nous n'avons à priori aucune connaissance de ces valeurs, pour certains cas l'initialisation sera trop éloignée de la solution optimale et l'algorithme divergera.

2.3.Méthodes fonctionnelles basées sur des linéarisations successives.

2.3.1. Méthode de quasilinearisation:

Dérivée de la méthode de Newton, la quasilinearisation a été appliquée dès les années 60 sur des exemples de transfert plan terre-mars.(Mc Gill & Kenneth 1964, McCue 1967, Tapley & Lewallen 1967).

Cette méthode dont on montre la convergence quadratique (Mc Gill & Kenneth 1966, Leondes & Paines 1968) présente l'avantage de converger rapidement vers la solution optimale. Ainsi le faible nombre d'itérations compense le coût numérique relativement important de chaque itération.

Supposons que l'on dispose d'une commande et de la trajectoire nominale correspondante qui ne soient pas trop éloignées de la solution optimale. Considérons le système obtenu en remplaçant, dans les équations d'état et les équations adjointes, la commande par une commande $\vec{u}^*(\vec{x}, \vec{\lambda})$ qui minimise le hamiltonien étant donné l'état et l'adjoint nominaux. A priori, les conditions finales et les équations différentielles de l'état et de l'adjoint ne seront pas vérifiées avec cette nouvelle commande. Les variations de l'état, de l'adjoint et éventuellement du multiplicateur de Lagrange seront alors calculés en résolvant un système linéarisé autour de la trajectoire nominale:

$$d\left(\dot{\vec{x}} - \vec{f}^*\right) + \dot{\vec{x}} - \vec{f}^* = 0 \quad (2.33)$$

$$d\left(\dot{\vec{\lambda}} - \vec{h}^*\right) + \dot{\vec{\lambda}} - \vec{h}^* = 0 \quad (2.34)$$

$$d\left(\vec{x}(t_0) - \vec{x}_0\right) + \vec{x}(t_0) - \vec{x}_0 = 0 \quad (2.35)$$

$$d\left(\vec{\psi}\right) + \vec{\psi} = 0 \quad (2.36)$$

$$d\left(\vec{\lambda}^T(t_f) - \frac{\partial \Phi}{\partial \vec{x}}\right) + \vec{\lambda}^T(t_f) - \frac{\partial \Phi}{\partial \vec{x}} = 0 \quad (2.37)$$

où \vec{f}^* et \vec{h}^* sont les seconds membres des équations différentielles état et adjoint dans lesquels la commande est remplacée par $\vec{u}^*(\vec{x}, \vec{\lambda})$.

Cette méthode permet d'atteindre au moins un minimum local du Hamiltonien. Elle semble donc très intéressante pour accélérer la fin de convergence d'une méthode du premier ordre ou dans le cas où l'on dispose de fonctions initiales au voisinage de l'optimum.

La description détaillée de la quasilinearisation fera l'objet du chapitre 2.

2.3.2. Pseudolinéarisation et linéarisation par difféomorphisme

Dans le cadre de la linéarisation, nous présentons ici la pseudolinéarisation (Reboulet & Champetier 1984, Reboulet et al. 1986) et la linéarisation par difféomorphisme (Claude 1984). Sans être des méthodes d'optimisation, ce type de linéarisation peut se révéler très utile dans le cadre d'un problème de régulation optimale.

La linéarisation par immersion et difféomorphisme local consiste à transformer une dynamique linéaire par rapport à la commande du type

$$\dot{\vec{x}} = \vec{f}(\vec{x}) + \mathbf{G}(\vec{x}) \cdot \vec{v} \quad (2.38)$$

en une dynamique linéaire contrôlable mise sous la forme:

$$\dot{\vec{z}} = \mathbf{F} \cdot \vec{z} + \mathbf{K} \cdot \vec{u} \quad (2.39)$$

où F et K se décomposent par bloc de la façon suivante:

$$F = (F_{ij}) \quad K = (K_i)$$

$$F_{ii} = \begin{pmatrix} 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 1 & \\ \dots & \dots & \dots & \dots \\ x & x & \dots & x \end{pmatrix} \quad F_{ij} = \begin{pmatrix} 0 & \dots & 0 \\ \vdots & \vdots & \vdots \\ 0 & \dots & 0 \\ \dots & \dots & \dots \\ x & x & \dots & x \end{pmatrix}$$

$$K_i = \begin{pmatrix} 0 & \dots & \dots & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & 1 & & \\ \dots & \dots & \dots & \dots & \dots \\ x & \dots & x & \dots & x \end{pmatrix} \quad \text{avec } i, j = 1 \dots n$$

F_{ii} est de dimension $\kappa_i \times \kappa_i$, F_{ij} de dimension $\kappa_i \times \kappa_j$ et K_i de dimension $\kappa_i \times n$, les x désignent des termes non déterminés à priori.

La pseudolinéarisation consiste à trouver une transformation non linéaire du type:

$$z_i = T_i(\vec{x}) \quad i=1, nx \quad (2.40)$$

$$v_i = T_{n+i}(\vec{x}, \vec{u}) \quad i=1, nu \quad (2.41)$$

telle que le système linéaire tangent pour le système transformé ne dépend plus du point de fonctionnement. Cette transformation permet d'avoir des systèmes linéaires tangents indépendants du point de fonctionnement et pourra être très utile dans le cas où on étudie ces systèmes au voisinage d'un point d'équilibre. Cependant le critère devient d'une grande complexité. De plus, il n'est pas certain qu'il existe une bijection entre l'espace initial et l'espace linéarisé.

2.4. Calcul d'une commande par retour d'état

2.4.1. Méthode de programmation dynamique différentielle

Le principe de cette méthode, basée sur la théorie de la programmation dynamique et les équations de Hamilton-Bellman-Jacobi, consiste à approcher par un développement au premier ou au second ordre, la fonction de cout V autour d'une trajectoire nominale.

Les termes du développement seront déterminés par intégration à rebours d'équations adjointes et les variations seront calculées de façon à vérifier les

hypothèses du développement. Nous présentons ici la méthode du premier ordre pour un problème sans conditions finales.

Si nous considérons une commande nominale $\vec{u}(t)$, la trajectoire correspondante $\vec{x}(t)$ et le cout correspondant V^0 . Une variation au premier ordre de $\vec{x}(t)$ et de $\vec{u}(t)$ implique

$$-\frac{\partial V}{\partial t} - \frac{\partial}{\partial t} \left(\frac{\partial V}{\partial \vec{x}} \right) \delta \vec{x} \approx \min_{\delta \vec{u}} \left\{ L(\vec{x} + \delta \vec{x}, \vec{u} + \delta \vec{u}, t) + \left(\frac{\partial V}{\partial \vec{x}} + \frac{\partial^2 V}{\partial \vec{x}^2} \delta \vec{x} \right) \cdot \vec{f}(\vec{x} + \delta \vec{x}, \vec{u} + \delta \vec{u}, t) \right\} \quad (2.43)$$

soit

$$-\frac{\partial V}{\partial t} - \frac{\partial}{\partial t} \left(\frac{\partial V}{\partial \vec{x}} \right) \delta \vec{x} \approx \min_{\delta \vec{u}} \left\{ H(\vec{x} + \delta \vec{x}, \vec{u} + \delta \vec{u}, t) + \delta \vec{x}^T \cdot \frac{\partial^2 V}{\partial \vec{x}^2} \cdot \vec{f}(\vec{x} + \delta \vec{x}, \vec{u} + \delta \vec{u}, t) \right\} \quad (2.44)$$

Considérons la commande \vec{u} telle que

$$\hat{u} = \operatorname{argmin}_{\vec{u}} H(\vec{x}, \frac{\partial V}{\partial \vec{x}}, \vec{u}, t) \quad (2.45)$$

Mesurons maintenant la variation de commande non plus par rapport à \vec{u} nominal mais par rapport à \hat{u} et supposons que cette variation reste du premier ordre. L'équation précédente devient (en remarquant que $\frac{\partial H}{\partial \vec{u}} = 0$)

$$-\frac{\partial V}{\partial t} - \frac{\partial}{\partial t} \left(\frac{\partial V}{\partial \vec{x}} \right) \delta \vec{x} \approx \min_{\delta \vec{u}} \left\{ H(\vec{x}, \hat{u}, t) + \frac{\partial H}{\partial \vec{x}} \delta \vec{x} + \delta \vec{x}^T \cdot \frac{\partial^2 V}{\partial \vec{x}^2} \cdot \vec{f}(\vec{x}, \hat{u}, t) \right\} \quad (2.46)$$

Cette équation étant vraie pour toute variation de l'état du premier ordre, nous avons les relations suivantes

$$-\frac{\partial V}{\partial t}(\vec{x}, t) = H(\vec{x}, \hat{u}, t) \quad (2.47)$$

$$-\frac{\partial}{\partial t} \left(\frac{\partial V}{\partial \vec{x}} \right) = \frac{\partial H}{\partial \vec{x}} + \frac{\partial^2 V}{\partial \vec{x}^2} \cdot \vec{f}(\vec{x}, \hat{u}, t) \quad (2.48)$$

Si nous définissons la fonction

$$a = V(\vec{x}, t) - V^o(\vec{x}, t) \quad (2.49)$$

où V est le coût obtenu avec la commande optimale $\vec{u}^*(t)$ et V^o le coût obtenu avec la commande nominale, la valeur en t_0 de cette fonction donne la variation de l'indice de performance que l'on veut obtenir : ΔV désirée = $V(\vec{x}_0, t_0) - V^o(\vec{x}_0, t_0)$

$$\Delta V \text{ désiré} = a(t_0)$$

En remarquant que

$$\begin{aligned} \frac{\partial V}{\partial t} &= (\overset{\circ}{a} + V^o) - \frac{\partial V}{\partial \vec{x}} \cdot \vec{x} \\ &= (\overset{\circ}{a} - L) - \frac{\partial V}{\partial \vec{x}} \cdot \vec{f} \end{aligned} \quad (2.50)$$

$$\frac{d}{dt} \left(\frac{\partial V}{\partial \vec{x}} \right) = \frac{\partial}{\partial t} \left(\frac{\partial V}{\partial \vec{x}} \right) + \frac{\partial^2 V}{\partial \vec{x}^2} \cdot \vec{f}(\vec{x}, \vec{u}, t) \quad (2.51)$$

les relations obtenues précédemment s'écrivent

$$\overset{\circ}{a} = - \left(H(\vec{x}, \frac{\partial V}{\partial \vec{x}}, \vec{u}, t) - H(\vec{x}, \frac{\partial V}{\partial \vec{x}}, \vec{u}, t) \right) \quad (2.52)$$

$$\begin{aligned} - \frac{\partial}{\partial t} \left(\frac{\partial V}{\partial \vec{x}} \right) &= \frac{\partial H}{\partial \vec{x}} + \frac{\partial^2 V}{\partial \vec{x}^2} \cdot \left(\vec{f}(\vec{x}, \vec{u}, t) - \vec{f}(\vec{x}, \vec{u}, t) \right) \\ &= \frac{\partial H}{\partial \vec{x}}(\vec{x}, \frac{\partial V}{\partial \vec{x}}, \vec{u}, t) \quad (\text{au premier ordre}) \end{aligned} \quad (2.53)$$

Ainsi en intégrant ces équations à rebours à partir des conditions finales

$$a(t_f) = 0 \quad (2.54)$$

$$\frac{\partial V}{\partial \vec{x}}(t_f) = \varphi(\vec{x}_f, t_f) \quad (2.55)$$

nous obtenons l'adjoint et une nouvelle commande \vec{u} qui permettent théoriquement d'améliorer le coût de $a(t_0) = V(\vec{x}_0, t_0) - V^o(\vec{x}_0, t_0)$. Les hypothèses de variation du premier ordre impliquent que la variation donnée par $a(t_0)$ doit correspondre à la variation réelle ΔV obtenue avec la nouvelle commande. Dans le cas où ΔV et $a(t_0)$ sont très différents, il faudra restreindre les variations. Le processus est réitéré tant que la variation du coût reste supérieure à une valeur donnée.

Remarquons que cette méthode ne traite pas les contraintes courantes. Seules les contraintes du type $g(u,t) \leq 0$ sont prises en compte dans une modification de la méthode proposée par Jacobson (1968).

2.4.2.Méthodes utilisant les crochets de Lie

Cette approche (Belghith & Rosset 1984, Bourdache-Siguerdidjane 1987) est basée sur l'élimination de l'adjoint à partir des dérivées successives de l'Hamiltonien par rapport au temps calculées au moyen de la théorie des crochets de Lie. Cette méthode concerne les problèmes sans contraintes, à temps final fixé et où la commande est de même dimension que l'état. Elle s'appuie sur le résultat suivant:

Pour les problèmes définis précédemment, on montre que la loi de commande optimale en boucle fermée vérifie une équation aux dérivées partielles du premier ordre du type:

$$\begin{aligned}
 C(\vec{x}, \vec{u}^*, t) \cdot \left(\frac{\partial \vec{u}^*}{\partial t} + \frac{\partial \vec{u}^*}{\partial \vec{x}} \cdot \vec{f} \right) &= D(\vec{x}, \vec{u}^*, t) \quad (2.56) \\
 C(\vec{x}, \vec{u}^*, t) &= \frac{\partial^2 L}{\partial \vec{u}^2} - \frac{\partial^2 \vec{f}}{\partial \vec{u}^2} \left(\mathbf{I}_{nx} \otimes \left(\frac{\partial \vec{f}}{\partial \vec{u}} \right)^{-T} \frac{\partial L^T}{\partial \vec{u}} \right) \\
 D(\vec{x}, \vec{u}^*, t) &= \frac{\partial^2 \vec{f}^T}{\partial \vec{u} \partial t} \cdot \left(\frac{\partial \vec{f}}{\partial \vec{u}} \right)^{-T} \cdot \frac{\partial L^T}{\partial \vec{u}} - \frac{\partial^2 L}{\partial \vec{u} \partial t} + \frac{\partial^2 \vec{f}^T}{\partial \vec{u} \partial \vec{x}} \cdot \left(\mathbf{I}_{nx} \otimes \left(\frac{\partial \vec{f}}{\partial \vec{u}} \right)^{-T} \frac{\partial L^T}{\partial \vec{u}} \right) \cdot \vec{f} \\
 &\quad - \frac{\partial^2 L}{\partial \vec{u} \partial \vec{x}} \cdot \vec{f} + \frac{\partial \vec{f}^T}{\partial \vec{u}} \cdot \frac{\partial L^T}{\partial \vec{x}} - \frac{\partial \vec{f}^T}{\partial \vec{u}} \cdot \frac{\partial \vec{f}^T}{\partial \vec{x}} \cdot \left(\frac{\partial \vec{f}}{\partial \vec{u}} \right)^{-T} \cdot \frac{\partial L^T}{\partial \vec{u}}
 \end{aligned}$$

Cette équation permet de calculer directement la commande optimale par retour d'état. En particulier dans le cas où le coût ne dépend pas directement de la commande, on obtient des équations algébriques.

Le résultat précédent peut être retrouvé à partir de l'équation de Hamilton-Bellman-Jacobi. La commande optimale en boucle fermée vérifie le système :

$$\frac{\partial V}{\partial t} + \frac{\partial V}{\partial \vec{x}} \cdot \vec{f} + L = 0 \quad (2.57)$$

$$\frac{\partial V}{\partial \vec{x}} \cdot \frac{\partial \vec{f}}{\partial \vec{u}} + \frac{\partial L}{\partial \vec{u}} = 0 \quad (2.58)$$

En différentiant l'équation (2.57) par rapport à l'état et l'équation (2.58) par rapport à l'état et au temps, on a :

$$\frac{\partial^2 \mathbf{V}}{\partial t \partial \vec{x}} + \vec{f}^T \cdot \frac{\partial^2 \mathbf{V}}{\partial \vec{x}^2} + \frac{\partial \mathbf{V}}{\partial \vec{x}} \cdot \left(\frac{\partial \vec{f}}{\partial \vec{x}} + \frac{\partial \vec{f}}{\partial \vec{u}} \cdot \frac{\partial \vec{u}}{\partial \vec{x}} \right) + \frac{\partial \mathbf{L}}{\partial \vec{x}} = 0 \quad (2.59)$$

$$\frac{\partial \vec{f}^T}{\partial \vec{u}} \cdot \frac{\partial^2 \mathbf{V}}{\partial \vec{x}^2} + \frac{\partial \mathbf{V}}{\partial \vec{x}} \left(\frac{\partial^2 \vec{f}}{\partial \vec{u}^2} \cdot \frac{\partial \vec{u}}{\partial \vec{x}} + \frac{\partial^2 \vec{f}}{\partial \vec{u} \partial \vec{x}} \right) + \frac{\partial^2 \mathbf{L}}{\partial \vec{u} \partial \vec{x}} = 0 \quad (2.60)$$

$$\frac{\partial \vec{f}^T}{\partial \vec{u}} \cdot \frac{\partial^2 \mathbf{V}}{\partial \vec{x} \partial t} + \frac{\partial \mathbf{V}}{\partial \vec{x}} \left(\frac{\partial^2 \vec{f}}{\partial \vec{u}^2} \cdot \frac{\partial \vec{u}}{\partial t} + \frac{\partial^2 \vec{f}}{\partial \vec{u} \partial t} \right) + \frac{\partial^2 \mathbf{L}}{\partial \vec{u} \partial t} = 0 \quad (2.61)$$

Le résultat (2.36) sera obtenu en éliminant de ces équations les termes $\frac{\partial^2 \mathbf{V}}{\partial t \partial \vec{x}}$, $\frac{\partial \mathbf{V}}{\partial \vec{x}}$ et $\frac{\partial^2 \mathbf{V}}{\partial \vec{x}^2}$.

2.4.3. Solution approchées par développement en série

Pour certains systèmes, des méthodes ont été développées en utilisant une approche par développement en séries. Nous citerons les méthodes de Willemstein (1977), de Rotella et Dauphin-Tanguy (1988) et de Mishne (1990).

Willemstein (1977) étudie une problème de commande optimale où les fonctions \vec{f} , \mathbf{L} et φ peuvent être développées en séries de puissances. Il se place au voisinage d'un point fixe autour duquel ces fonctions peuvent s'écrire sous la forme

$$\vec{f} = \mathbf{A} \cdot \vec{x} + \mathbf{B} \cdot \vec{u} + \vec{F}(\vec{x}, \vec{u}) \quad (2.62)$$

$$\mathbf{L} = \vec{x}^T \cdot \mathbf{Q} \cdot \vec{x} + \vec{u}^T \cdot \mathbf{R} \cdot \vec{u} + l(\vec{x}, \vec{u}) \quad (2.63)$$

$$\varphi = \vec{x}_f^T \cdot \mathbf{M} \cdot \vec{x}_f + \mu(\vec{x}_f, t_f) \quad (2.64)$$

où \vec{F} , l et μ contiennent des termes d'ordre supérieur en \vec{x} et \vec{u} .

Avec comme hypothèses : $\mathbf{Q} \geq 0$ et $\mathbf{R} > 0$, il montre que la commande et le coût optimaux peuvent être développés sous la forme d'une série de puissances :

$$\vec{u}^* = \vec{u}^{*(1)}(\vec{x}, t) + \vec{u}^{*(2)}(\vec{x}, t) + \vec{u}^{*(3)}(\vec{x}, t) + \dots$$

$$J = J^{(2)}(\vec{x}, t) + J^{(3)}(\vec{x}, t) + J^{(4)}(\vec{x}, t) + \dots$$

où $\vec{u}^{*(i)}(\vec{x}, t)$ et $J^{(i)}$ sont les termes d'ordre i en \vec{x} et $\vec{u}^{*(1)}(\vec{x}, t)$ et $J^{(2)}(\vec{x}, t)$ les solutions du problème linéaire quadratique approché.

$$\vec{u}^{*(1)}(\vec{x}, t) = -\mathbf{R}^{-1} \mathbf{B}^T \mathbf{K} \cdot \vec{x} \quad (2.65)$$

$$J^{(2)}(\vec{x}, t) = \vec{x}^T \cdot \mathbf{K} \cdot \vec{x} \quad (2.66)$$

La matrice \mathbf{K} est la solution de l'équation différentielle de RICCATI . Les termes d'ordre supérieur pourront être calculés de façon récursive. Ces termes seront donnés par les équations

$$\begin{aligned} J_x^{(m)} \cdot \mathbf{A} \cdot \vec{x} + J_t^{(m)} = & - \sum_{k=3}^{m-1} J_x^{(k)} \mathbf{B} \vec{u}^{*(m-k-1)} \\ & - \sum_{k=2}^{m-1} J_x^{(k)} F^{(m-k-1)} \\ & - 2 \sum_{k=2}^{(m-1)/2} \vec{u}^{*(k)T} \mathbf{R} \vec{u}^{*(m-k)} \\ & - \vec{u}^{*(m/2)T} \mathbf{R} \vec{u}^{*(m/2)} - l^{(m)} \quad m=3,4,\dots \end{aligned} \quad (2.67)$$

$$\vec{u}^{*(k)} = -\frac{1}{2} \mathbf{R}^{-1} \cdot \left(\mathbf{B}^T \cdot J_x^{(k+1)T} + \sum_{i=1}^{k-1} F_u^{(i)T} \cdot J_x^{(k-i+1)T} + l_u^{(m)T} \right) \quad k=2,3,\dots \quad (2.68)$$

Pour plus de clarté, nous avons adopté ici la notation indicielle pour les dérivées partielles.

Une autre approche (Rotella & Dauphin-Tanguy 1988) utilise la notation des tenseurs et les produits de Kronecker pour construire de façon récursive une commande optimale par retour d'état de certains systèmes. Cette méthode s'applique aux systèmes du type

$$\dot{\vec{x}} = \vec{f}(\vec{x}, t) + \mathbf{G}(\vec{x}, t) \cdot \vec{u} \quad (2.69)$$

où \vec{f} et \mathbf{G} sont analytiques en \vec{x} . Le critère à minimiser est le suivant

$$J = \frac{1}{2} \int_{t_0}^{\infty} \vec{s}^T \cdot \mathbf{Q} \cdot \vec{s} + \vec{u}^T \cdot \mathbf{R} \cdot \vec{u} \, dt \quad (2.70)$$

où \vec{s} est un vecteur de sortie, fonction analytique de \vec{x} : $\vec{s} = \vec{h}(\vec{x}, t)$.

La méthode permet d'obtenir une expression analytique de la commande sous la forme d'une série infinie:

$$\vec{u}^*(\vec{x}, t) = \sum_{k=1}^{\infty} U_k \cdot \vec{x}^{\rightarrow|k|} \quad (2.71)$$

où U_k sont des matrices de dimension appropriées et $\vec{x}^{\rightarrow|k|}$ la k ème puissance de \vec{x} au sens des produits de Kronecker :

$$\vec{x}^{\rightarrow|0|} = 1 \quad (2.72)$$

$$\vec{x}^{\rightarrow|k|} = \vec{x}^{\rightarrow|k-1|} \otimes \vec{x} = \vec{x}^{\rightarrow|k-1|} \otimes \vec{x} \quad (2.73)$$

Par rapport à la méthode de Willemstein ou d'autres méthodes basées sur les calculs de variations, cette méthode a l'avantage de donner une expression analytique non linéaire de la commande par retour d'état . En pratique, la commande sous optimale sera obtenue en tronquant la série à l'ordre m . Bien que le choix de m ne soit pas un problème trivial, cette technique permet d'obtenir une expression simple d'une commande sous optimale en boucle fermée.

Mishne (1990) propose une méthode de résolution pour des systèmes particuliers qui peuvent se mettre sous la forme

$$\dot{\vec{x}} = \vec{f}(\vec{x}, \vec{u}, t) + \epsilon \cdot \vec{g}(\vec{x}) \quad (2.74)$$

Le problème d'optimisation considéré consiste à minimiser un critère final $J = \varphi(\vec{x}_f, t_f)$, sous la contrainte finale $\vec{\psi}(\vec{x}_f, t_f) = 0$. Cette méthode est basée sur des développements en ϵ et suppose que l'équation de Bellman-Hamilton-Jacobi admet une solution analytique pour $\epsilon = 0$. La commande optimale, le coût correspondant et l'adjoint sont développés en une somme de puissances de ϵ

$$V(\vec{x}, t) = \sum_{i=0}^{\infty} V(\vec{x}, t)_{(i)} \cdot \epsilon^i \quad (2.75)$$

$$\vec{u}^*(t) = \sum_{i=0}^{\infty} \vec{u}^*_{(i)}(t) \cdot \epsilon^i \quad (2.76)$$

$$\vec{\lambda}(t) = \sum_{i=0}^{\infty} \frac{\partial V_{(i)}}{\partial \vec{x}} \cdot \epsilon^i \quad (2.77)$$

En remplaçant $\vec{f}(\vec{x}, \vec{u}^*, t)$ par son développement autour de $\vec{u}^*_{(0)}(t)$ et en rassemblant les termes du même ordre en ϵ , on obtient des équations récursives qui permettent de calculer la solution optimale à partir de la solution d'ordre zéro obtenue en résolvant le problème pour $\epsilon = 0$. Les termes d'ordre supérieur seront directement obtenus par:

$$V_{(i)} = - \int_t^{t_f} R_i \cdot dt \quad (2.78)$$

$$\lambda_{(i)} = - \int_t^{t_f} \frac{\partial R_i}{\partial \vec{x}} \cdot dt \quad (2.79)$$

où

$$R_i = - \frac{\partial V_{(0)}}{\partial \vec{x}} \cdot \frac{\partial \vec{f}}{\partial \vec{u}} \cdot \vec{u}_{(i)}^*(t) - \frac{\partial V_{(1)}}{\partial \vec{x}} \cdot \frac{\partial \vec{f}}{\partial \vec{u}} \cdot \vec{u}_{(i-1)}^*(t) - \dots - \dots - \frac{\partial V_{(i-1)}}{\partial \vec{x}} \cdot \frac{\partial \vec{f}}{\partial \vec{u}} \cdot \vec{u}_{(1)}^*(t) - \frac{\partial V_{(i-1)}}{\partial \vec{x}} \cdot \vec{g} \quad (2.80)$$

2.5.Méthodes paramétriques.

Les méthodes paramétriques consistent à simplifier le problème avant de le résoudre. Il s'agira de calculer une approximation de la solution optimale en remplaçant le problème d'optimisation fonctionnelle par un problème d'optimisation paramétrique.

Il existe principalement deux types de paramétrisation : soit la commande seule est paramétrisée, la trajectoire associée à cette commande approchée sera obtenue par intégration du système dynamique, soit la paramétrisation porte aussi sur l'état (et parfois l'adjoint) et dans ce cas, la trajectoire sera aussi une trajectoire approchée.

Notations : Dans cette partie nous noterons \vec{a} le vecteur des paramètres. Sa dimension sera na (le nombre de paramètres).

2.5.1. Paramétrisation de la commande

La première série de méthodes consiste à approcher la commande par une fonction dépendant du temps, d'un certain nombre de paramètres et éventuellement de l'état, c'est à dire que l'on pose:

$$\vec{u}(t) = \vec{u}(t, \vec{a}, \vec{x})$$

Le problème d'optimisation fonctionnelle est remplacé par le problème approché suivant

trouver le vecteur de paramètres \vec{a} qui minimise

$$J(\vec{a}) = \int_{t_0}^{t_f} L(\vec{x}(t), \vec{u}(t, \vec{a}, \vec{x}), t) dt + \varphi(\vec{x}(t_f), t_f)$$

tout en respectant les contraintes $M_i(\vec{a}) = 0 \quad i=1 \dots ncf$

$\vec{x}(t)$ étant défini par $\dot{\vec{x}} = \vec{F}(\vec{x}, \vec{u}(t, \vec{a}, \vec{x}), t)$ et $\vec{x}(t_0) = \vec{x}_0$

M_i par $M_i(\vec{a}) = \psi_i(\vec{x}(t_f), t_f)$

D'un point de vue théorique, Boudarel et al (1968), ont posé les bases d'une telle paramétrisation. D'un point de vue pratique, Williamson(1971), Hull (1975), Peterson et Rader (1981) proposent une commande de type polynomiale

$$\vec{u}(t) = \sum_{i=0}^N \vec{\alpha}_i \cdot t^i$$

\vec{a} sera le vecteur formé des coefficients $\alpha_{ij} \quad i=1 \dots N, \quad j=1 \dots nu$

Le problème d'optimisation paramétrique est résolu par une méthode du second ordre (Hull & Edgeman 1975). Rader et Hull (1975) résolvent un problème d'optimisation de trajectoire pour un avion de combat avec une commande approchée par une série de Chebychev d'ordre cinq.

Ce type d'approximation est cependant assez limité. Si la commande optimale est très non linéaire, il ne sera pas possible de l'approcher par des fonctions aussi simples et la solution obtenue sera très éloignée de l'optimale.

Pour obtenir plus de précision, une solution est de découper l'intervalle de temps en sous intervalles sur lesquels la commande sera paramétrisée par une fonction polynômiale par morceaux:

$$\vec{u}(t) = \sum_{i=0}^{np} \vec{\alpha}_{ij} \cdot t^i \quad \text{pour } t \in [t_j, t_{j+1}] \quad j=0 \dots M-1, \quad t_M = t_f$$

np étant le degré du polymôme. Sirisena (1973) utilise des polymômes de degré deux, Goh & Teo (1988) de degré zéro. Les paramètres à optimiser seront les coefficients des polymômes et les instants t_j ce qui donne le vecteur

$$\vec{a} = (\alpha_{ijk}, t_j) \quad i=1 \dots N, \quad j=1 \dots M, \quad k=1 \dots nu.$$

Ces paramètres seront calculés par des méthodes d'optimisation classiques comme par exemple celle Davidon-Fletcher-Powell (Sirisena & Tan 1974) ou de Goldfarb-lapidus (Sirisena & Chou 1976).

Cette méthode permet de mieux approcher la commande mais le nombre de paramètres peut devenir assez important selon la complexité de la commande optimale. En effet, il sera alors nécessaire d'augmenter le nombre de sous intervalles ou l'ordre des polynômes si les polynômes choisis sont trop simples (d'ordre zéro par exemple). Le nombre de paramètres est alors de

$$na = (np+1).M.nu$$

où np est l'ordre des polynômes et M le nombre de sous intervalles.

Le problème de ces méthodes réside dans le choix du nombre de paramètres. En effet, avoir une approximation assez précise de la commande demande un très grand nombre de paramètres, nous perdons ainsi le gain en temps de calcul. A l'inverse, trop simplifier la forme de la commande nous éloigne de la solution optimale. Il s'agit de trouver le bon compromis, ce qui dépend beaucoup de l'exemple traité.

Pour tenir compte des connaissances que l'on peut avoir de façon empirique sur le système étudié, Blank & Shinar (1978) proposent de définir sur chaque sous intervalle $[t_i; t_{i+1}]$ une commande du type

$$\vec{u} = \vec{u}(t, \vec{x}, \vec{a})$$

Menon et al (1987) utilisent cette méthode avec une commande par retour dépendant de quelques paramètres dans un problème d'optimisation d'une trajectoire de missile. Ils proposent une commande du type

$$\vec{u} = \vec{u}(t, \vec{x}, \vec{k}_r, \vec{x}_r)$$

où \vec{k}_r et \vec{x}_r sont des fonctions constantes par morceaux, les paramètres à optimiser seront les caractéristiques de ces fonctions.

Ce type de commande permet de définir une commande par retour, forme particulièrement adaptée pour les problèmes de guidage optimal. Cependant la forme de la commande sera spécifique au problème traité et l'efficacité de l'approximation dépendra beaucoup de l'exactitude du modèle de commande choisi.

2.5.2. Paramétrisation de l'état, de l'adjoint et/ou de la commande

La seconde série de méthodes consiste à approcher l'état, l'adjoint et éventuellement la commande par des fonctions plus simples définies par des paramètres. Les équations différentielles induisent alors des conditions sur ces paramètres et le problème consistera à minimiser le critère en respectant ces conditions ainsi que les contraintes du problème initial.

Le problème approché sera de trouver un vecteur \vec{a} tel que

le critère J soit minimum
 les contraintes soient respectées
 l'erreur sur $\dot{\vec{x}} - \vec{f}$ soit faible

avec

$$\begin{aligned}\vec{u}(t) &= \vec{u}(\vec{a}, t) \\ \vec{x}(t) &= \vec{x}(\vec{a}, t)\end{aligned}$$

Mehra & Davis (1972) utilisent une approximation aux différences finies sur les intervalles $[t_i, t_{i+1}]$ en posant

$$\vec{x}_{i+1} - \vec{x}_i / (t_{i+1} - t_i) = \vec{f}(\vec{x}_i, \vec{u}_i) \quad (2.81)$$

Hargraves & Paris (1987), Ilgenfritz & Gailey (1988) et Betts(1990) approximent l'état sur chacun des sous-intervalles par un polynôme d'ordre trois déterminé par

$$\vec{x}_{i+1}, \vec{x}_i, f(\vec{x}_i, \vec{u}_i) \text{ et } f(\vec{x}_{i+1}, \vec{u}_{i+1})$$

L'état sera donné par

$$\vec{x}(t) = \vec{P}_i(t) \quad \forall t \in [t_i; t_{i+1}] \quad (2.82)$$

$$\text{avec } \vec{P}_i(t_i) = \vec{x}_i$$

$$\vec{P}_i(t_{i+1}) = \vec{x}_{i+1} \quad (2.83)$$

$$\dot{\vec{P}}_i(t_i) = f(\vec{x}_i, \vec{u}_i) \quad (2.84)$$

$$\dot{\vec{P}}_i(t_{i+1}) = f(\vec{x}_{i+1}, \vec{u}_{i+1}) \quad (2.85)$$

Pour minimiser les erreurs sur la dynamique, ils définissent la contrainte suivante

$$\vec{P}_i(t_m) - f(\vec{x}_m, \vec{u}_m) = 0$$

où t_m est le milieu du sous intervalle

$$\vec{x}_m = \vec{P}_i(t_m)$$

\vec{u}_m sera soit un paramètre à optimiser au même titre que \vec{x}_i et \vec{u}_i

soit une approximation linéaire entre \vec{u}_i et \vec{u}_{i+1}

Il s'agira donc de minimiser $J(\vec{a})$ avec

$$\vec{a} = (\vec{u}_0, \vec{x}_1, \vec{u}_1, \vec{x}_2, \vec{u}_2, \dots, \vec{x}_f, \vec{u}_f)$$

(avec éventuellement les \vec{u}_{mi})

Suivant le même principe, Hargraves, Johnson, Paris & Rettie (1979) approchent l'état par un polynôme de Chebychev tandis que Sheela & Ramamoorthy (1981) choisissent une approximation de Raleigh-ritz. Les applications sont multiples : montée en temps minimum, transfert d'orbite, interception, évacion optimale dans un cas de poursuite...

Une autre transformation du problème, consiste à se placer dans un espace de fonctions d'approximation. L'état et la commande seront approchés par une série d'ordre N et, en utilisant les opérateurs sur cet espace, les équations différentielles seront remplacées par des équations algébriques sur les termes des séries. Il faudra alors minimiser

$$P(\vec{a}) \text{ avec } G(\vec{a}) = 0$$

Ces méthodes ont été largement développées dans le domaine linéaire : Stavroulakis & Tzafestas (1977) et Chen & al (1978) utilisent des séries de Walsh, Hsu & Heng (1983) des fonctions constantes par morceaux et Paraskevopoulos(1983), des séries de Chebychev. Betts (1988), Russel & Shampine (1971), Vlassenbroek & Van dooren (1988) généralisent cette méthode aux systèmes non linéaires. Ils posent

$$\vec{u}(t) = \sum_{i=1}^N \vec{\alpha}_i \cdot \zeta_i(t) \quad (2.86)$$

$$\vec{x}(t) = \sum_{i=1}^N \vec{\beta}_i \cdot \zeta_i(t) \quad (2.87)$$

où les second membres représentent les séries de chebychev à l'ordre N.

En utilisant les propriétés des polynômes de Chebychev, l'indice de performance deviendra

$$J = \gamma_0(\vec{\alpha}, t_f) + \sum_{n=1}^M \gamma_n(\vec{\alpha}, t_f) + \delta_0(\vec{\alpha}, \vec{\beta}, t_f) - \sum_{n=1}^M \frac{1+(-1)^n}{n^2-1} \delta_n(\vec{\beta}, \vec{\alpha}, t_f) \quad (2.88)$$

où les γ_n viennent de la série approchant φ et les δ_n de la série approchant L

γ_n et δ_n seront donnés par:

$$\gamma_n(\vec{\alpha}, t_f) = \frac{2}{K} \sum_{i=1}^K \varphi(\vec{x}(\cos\theta_i), t_f) \cdot \cos n\theta_i \quad (2.89)$$

$$\delta_n(\vec{\beta}, \vec{\alpha}, t_f) = \frac{2}{K} \sum_{i=1}^K L(\vec{x}(\cos\theta_i), \vec{u}(\cos\theta_i), t_f) \cdot \cos n\theta_i \quad (2.90)$$

$$n = 1 \dots M \quad K > M$$

$$\theta_i = \frac{2i-1}{K} \cdot \frac{\pi}{2} \quad (2.91)$$

et l'équation dynamique sera remplacée par

$$\dot{\vec{x}}(t) = \frac{1}{2} \vec{\pi}_0(\vec{\beta}, \vec{\alpha}, t) \cdot \zeta_0(t) + \sum_{j=1}^P \vec{\pi}_j(\vec{\beta}, \vec{\alpha}, t) \cdot \zeta_j(t) \quad (2.92)$$

où $\vec{\pi}_i$ vient de l'approximation de \vec{F}

$$\vec{\pi}_i(\vec{\beta}, \vec{\alpha}, t) = \frac{2}{K} \sum_{i=1}^K \vec{F}(\vec{x}(\cos\theta_i), \vec{u}(\cos\theta_i), t_f) \cdot \cos n\theta_i \quad (2.93)$$

$$n = 1 \dots P \quad K > P$$

$$\theta_i = \frac{2i-1}{K} \cdot \frac{\pi}{2} \quad (2.94)$$

Cette dernière équation se traduit, en utilisant l'opérateur dérivée sur les séries, par des relations entre les coefficients $\vec{\beta}$ et $\vec{\alpha}$ du type

$$F_n(\vec{\beta}, \vec{\alpha}, t) = 0 \quad \text{pour } n = 1 \dots P+2$$

Par conséquent le problème paramétrique est

$$\text{minimiser } J(\vec{\beta}, \vec{\alpha})$$

$$\text{avec } F_n(\vec{\beta}, \vec{\alpha}, t) = 0 \quad \text{pour } n = 1 \dots P+2$$

Tout le problème de cette méthode sera d'évaluer jusqu'à quel ordre N, M et P développer les séries pour que l'approximation soit assez précise sans augmenter de façon importante la dimension des paramètres.

CONCLUSION: SYNTHÈSE COMPARATIVE DES DIFFÉRENTES MÉTHODES

Comme nous pouvons le constater après cette revue des méthodes d'optimisation, les approches sont très nombreuses et sont basées parfois sur des philosophies très différentes. Chacune de ces approches présente des qualités mais aussi des limites. Bien qu'il existe quelques études comparatives publiées antérieurement (Tapley et al. 1967, Blank et Shinar 1982) nous n'avons pas trouvé dans la littérature de revue complète récente sur les méthodes de calcul de commande optimale.

Nous nous proposons d'effectuer une synthèse comparative de toutes ces méthodes. Le but de cette partie sera de donner certains repères afin de faciliter le choix d'une méthode pour un problème donné selon le critère que l'on privilégie. Les critères que nous avons examinés sont les suivants

- simplicité de formulation
- capacité mémoire requise
- domaine de convergence
- vitesse de convergence
- généralité de la méthode et traitement des contraintes
- possibilité de calcul en boucle fermée.

Les méthodes du gradient simple présentent l'avantage d'avoir une grande robustesse vis à vis de l'initialisation. La formulation de la méthode est relativement simple mais la vitesse de convergence diminue beaucoup au voisinage de la solution optimale. Cette vitesse peut cependant être accélérée par différents moyens: par des méthodes de correction du gradient (du type méthode PARTAN en optimisation paramétrique), en utilisant la minimisation du hamiltonien ou en utilisant des méthodes du second ordre. Ces méthodes du second ordre ont une très bonne vitesse de convergence mais sont très sensibles à l'initialisation. Ces méthodes peuvent être associées en fin de convergence avec une méthode du premier ordre. L'encombrement mémoire reste relativement important.

La méthode des extrémaux voisins est relativement complexe à mettre en œuvre mais permet pour de petites perturbations autour d'une trajectoire optimale connue d'obtenir rapidement la nouvelle trajectoire optimale. Cette méthode reste limitée pour

le calcul d'une trajectoire optimale. Cependant la variation de la commande est obtenue en fonction des perturbations, cela permet d'envisager son utilisation pour le guidage dans la mesure où les perturbations par rapport à la trajectoire nominale sont faibles.

Les méthodes de tir sont très simples à formuler. De plus, lorsque les itérations portent non plus sur les fonctions mais sur les valeurs $\vec{\lambda}_0$ et \vec{v} l'algorithme demande peu de capacité mémoire. Elles sont cependant très sensibles à l'initialisation de l'adjoint. Par rapport aux méthodes du gradient, ces méthodes convergent plus rapidement.

Les méthodes de quasilinearisation convergent de façon quadratique. Bien que la formulation est relativement complexe et que le domaine de convergence soit relativement faible, ces méthodes assurent le calcul d'un optimum local assez rapidement.

Toutes ces méthodes sont très générales et traitent des problèmes soumis à des contraintes courantes et finales. La commande obtenue est en général une commande en boucle ouverte en dehors de la méthode de quasilinearisation qui donne une commande qui dépend de l'état mais aussi de l'adjoint et de la méthode de perturbation qui permet d'obtenir une variation de commande en fonction de la variation de l'état et des conditions finales.

Les méthodes par retour d'état ont des champs d'application restreint. En dehors de la programmation dynamique différentielle qui est une méthode itérative, elles sont toutes analytiques ce qui permet de calculer directement la solution exacte ou une approximation de la solution.

La programmation dynamique différentielle présente l'avantage d'être plus stable vis à vis de l'intégration du système adjoint. De plus cette méthode est simple à formuler. Sa vitesse de convergence se rapproche de celle des méthodes du second ordre. Cependant elle ne permet pas de traiter les contraintes de façon simple sauf celles du type $g(u) \leq 0$.

La méthode utilisant les crochets de Lie ne s'applique que pour les systèmes sans contraintes où l'état et la commande sont de même dimension. Son champ d'application reste donc relativement limité mais lorsque l'on peut l'utiliser, on obtient la solution exacte. La méthode de Willemstein s'applique aux systèmes développables en série de puissances, la méthode des produits de Kronecker aux systèmes linéaires

quadratiques par rapport à la commande et celle de Mishne à des systèmes particuliers où l'on fait apparaître un petit coefficient . Par ailleurs, ces méthodes ne traitent pas les contraintes courantes.

Les méthodes paramétriques peuvent être très simples à mettre en œuvre. Le volume de calcul est peu important et se résume même parfois à la simple résolution d'équations algébriques. Leur vitesse de convergence dépendra essentiellement de l'algorithme d'optimisation paramétrique utilisé. Le problème de ces méthodes sera le choix d'un type d'approximation et du nombre de paramètres. En effet ce choix conditionne la qualité et la précision de la solution obtenue. En dehors des paramétrisations du type $\vec{u} = \vec{u}(\vec{x}, \vec{a}, t)$, la commande sera une commande boucle ouverte.

Méthode	domaine de convergence	vitesse de convergence	encombrement mémoire	contraintes traitées	domaine d'application	
gradient 1	très large	faible en fin de convergence	important	contraintes finales et courantes	général	B.O.
gradient 2	faible	rapide	important	contraintes finales	général	B.O.
extrémaux voisins	très faible	rapide	moyen	contraintes finales et courantes	général	B.O.
methodes de tir	très faible	moyenne	faible	contraintes finales	général	B.O.
quasilinearisation	faible	rapide	important	contraintes finales et courantes	général	B.O.
programmation dynamique différentielle	faible	rapide	faible	$g(u) \leq 0$	général	B.F.
crochets de Lie	calcul analytique direct			aucune	$nu = nx$	B.F.
série de puissances	calcul récursif			aucune	$x = f(x,u)+e.g(x)$	B.F.
produits de Kronecker	calcul récursif			aucune	$x = f(x)+G(x).u$ critère quadratique	B.F.
paramétrisation de la commande	mêmes caractéristiques que la méthode d'optimisation paramétrique utilisée			toutes contraintes	général	B.F. possible B.O.
autres paramétrisation						

CHAPITRE 2

METHODE DE QUASILINEARISATION

CHAPITRE 2

METHODE DE QUASILINEARISATION

INTRODUCTION

Le propos de ce chapitre est d'une part de présenter de façon détaillée les méthodes de quasilinearisation, d'autre part de développer une nouvelle variante de la méthode. Cette méthode sera mise en œuvre afin de tester certaines caractéristiques sur des exemples pratiques.

Nous établissons tout d'abord l'historique de la méthode. Après avoir rappelé les résultats généraux relatifs au calcul de l'optimum de fonctions numériques à variables vectorielles par la méthode de Newton-Raphson, nous présentons les applications de cette méthode au calcul de la commande optimale. Les algorithmes développés dans la littérature à partir de ce principe se divisent en deux groupes: les méthodes de variations seconde et les méthodes de quasilinearisation. Nous recensons ces dernières et exposons l'état de l'art.

Dans un cadre général, nous développons une nouvelle variante de la quasilinearisation. Nous décrivons en détail chaque étape d'une itération de l'algorithme. Ces étapes sont principalement le calcul de variations et la résolution du système linéaire ainsi obtenu. Afin d'augmenter le domaine de convergence de l'algorithme, nous proposons une méthode originale de gestion des itérations. Cette méthode consiste à rajouter dans le calcul de variations des matrices de contraction afin de régler le pas de l'algorithme. Cet exposé est enfin complété par l'examen des diverses possibilités d'initialisation.

Toute cette analyse consiste en fait à valider cette variante de la quasilinearisation dans le but d'étudier son applicabilité dans un contexte de guidage. Afin de préciser les caractéristiques nécessaires pour cette application, nous détaillons le principe de ce guidage.

Un programme d'optimisation général est élaboré. Ce programme modulable permet de tester les différentes possibilités de la méthode exposée et d'analyser sur des exemples diversifiés le comportement de l'algorithme.

1. PRINCIPE DE LA METHODE

1.1. Méthode de Newton: résolution de $\vec{G}(\vec{z}) = 0$

La méthode de quasilinearisation est dérivée de la méthode de Newton. Rappelons que cet algorithme permet de calculer de façon itérative le zéro d'une fonction continûment différentiable $\vec{G}(\vec{z})$ de \mathbb{R}^n dans \mathbb{R}^n . Elle consiste à construire une suite définie par

$$\begin{aligned} \vec{z}^0 & \text{ donné} \\ \vec{G}(\vec{z}^n) + \frac{\partial \vec{G}}{\partial \vec{z}}(\vec{z}^n) \cdot d\vec{z} &= 0 \quad n=0,1,2,\dots \\ \vec{z}^{n+1} &= \vec{z}^n + d\vec{z} \end{aligned} \quad (1.1)$$

Kantorovich (1949) montre que si $(\partial \vec{G} / \partial \vec{z})^{-1}$ est définie et continue pour tout $\vec{z} \in B$ et si $\partial \vec{G} / \partial \vec{z}$ satisfait une condition de Lipschitz alors il existe des constantes strictement positives K_1 et K_2 telles que pour \vec{z}^0 dans un voisinage de \vec{z}^* , la suite (\vec{z}^n) définie par (2.1) et initialisée en \vec{z}^0 converge vers \vec{z}^* . On peut montrer que cette suite est telle que

$$\begin{aligned} \|\vec{z}^n - \vec{z}^*\|_B &\leq (K_1 / 2^n) \cdot d^{2^n} \\ d &= K_2 \cdot \|\vec{z}^1 - \vec{z}^0\|_B \end{aligned}$$

Ce théorème montre que pour \vec{z}^0 tel que $d < 1$, la suite converge plus "vite" que toute suite du type (Kp^n) où K est un réel strictement positif et $p \in [0, 1]$.

Considérons maintenant le problème qui consiste à trouver le minimum d'une fonction $J(\vec{z})$. En appliquant la méthode de Newton à la recherche du zéro de $\partial J / \partial \vec{z}$, l'algorithme permet de calculer un minimum local de J . La suite sera dans ce cas définie par

$$\begin{aligned} \frac{\partial J^T}{\partial \vec{z}}(\vec{z}^n) + \frac{\partial^2 J}{\partial \vec{z}^2}(\vec{z}^n) \cdot d\vec{z} &= 0 \quad n=0,1,2,\dots \\ \vec{z}^{n+1} &= \vec{z}^n + d\vec{z} \end{aligned} \quad (1.2)$$

Remarquons que \vec{z}^{n+1} peut aussi être interprété comme le résultat de la minimisation de l'approximation quadratique de la fonction J en \vec{z}^n :

$$J(\vec{z}) \approx J(\vec{z}^n) + \frac{\partial J}{\partial \vec{z}}(\vec{z}^n) \cdot (\vec{z} - \vec{z}^n) + \frac{1}{2} (\vec{z} - \vec{z}^n)^T \cdot \frac{\partial^2 J}{\partial \vec{z}^2}(\vec{z}^n) \cdot (\vec{z} - \vec{z}^n)$$

(1.3)

Un des premiers théorèmes de convergence de cette méthode est dû à Goldstein (1965). Il propose l'algorithme suivant

$$\vec{z}^{n+1} = \vec{z}^n - \varepsilon_n \cdot \left(\frac{\partial^2 J}{\partial \vec{z}^2}(\vec{z}^n) \right)^{-1} \cdot \frac{\partial J}{\partial \vec{z}}(\vec{z}^n) \quad (1.4)$$

où ε_n est tel que

$$\begin{aligned} - \varepsilon_n \cdot (1-\alpha) \cdot \frac{\partial J}{\partial \vec{z}}(\vec{z}^n) \cdot \left(\frac{\partial^2 J}{\partial \vec{z}^2}(\vec{z}^n) \right)^{-1} \cdot \frac{\partial J}{\partial \vec{z}}(\vec{z}^n) \\ \leq J(\vec{z}^{n+1}) - J(\vec{z}^n) \leq \\ - \varepsilon_n \cdot \alpha \cdot \frac{\partial J}{\partial \vec{z}}(\vec{z}^n) \cdot \left(\frac{\partial^2 J}{\partial \vec{z}^2}(\vec{z}^n) \right)^{-1} \cdot \frac{\partial J}{\partial \vec{z}}(\vec{z}^n) \end{aligned} \quad (1.5)$$

$$\alpha \in \left[0, \frac{1}{2}\right]$$

Il montre que si $\exists m, M$ constantes telles que $0 < m \leq M$ et

$$\forall \vec{y}, \vec{z} \quad m \cdot \|\vec{y}\|_B^2 \leq \vec{y}^T \cdot \left(\frac{\partial^2 J}{\partial \vec{z}^2}(\vec{z}) \right) \cdot \vec{y} \leq M \cdot \|\vec{y}\|_B^2$$

alors

(i) la suite ε_n est stationnaire égale à 1 à partir d'un certain rang

(ii) la suite \vec{z}^n est monotone décroissante

et converge vers le minimum local de J

En conclusion, l'algorithme de Newton permet sous certaines hypothèses, de trouver le minimum local d'une fonction réelle à variable vectorielle. Cet algorithme a une convergence quadratique. Cependant, il est nécessaire pour assurer la convergence, de l'initialiser par un vecteur qui se trouve au voisinage de la solution, ce qui veut dire que le domaine de convergence est relativement faible. Nous allons voir dans quelle mesure cet algorithme a été généralisé au problème de la commande optimale.

1.2. Opérateur de Newton-Raphson généralisé.

La méthode de Newton a tout d'abord été développée pour résoudre de façon générale un système d'équations différentielles non linéaires avec conditions aux deux bouts (Bellman et Kalaba 1965, Mc Gill et Kenneth 1964). Cette méthode est trouvée dans la littérature sous le nom d'opérateur de Newton-Raphson généralisée ou de quasilinearisation. La généralisation est la suivante. Considérons le système vectoriel différentiel:

$$\begin{aligned} \frac{d\vec{z}}{dt} - \vec{G}(\vec{z}, t) &= 0 \\ \vec{z}(t_1) &= \vec{z}_1 \\ \vec{z}(t_2) &= \vec{z}_2 \end{aligned} \quad (1.6)$$

Ici, \vec{z} est une fonction vectorielle de classe C^2 par rapport à une variable scalaire t et \vec{G} une fonction non linéaire de $\mathbb{R}^n \times \mathbb{R}$ dans \mathbb{R}^n .

De la même façon que pour le calcul du zéro d'une fonction vectorielle, on va calculer le zéro de l'opérateur $[d(\cdot)/dt - \vec{G}(\cdot, t)]$. On définit la suite de fonction (\vec{z}^n) par:

$$\begin{aligned} \vec{z}^0 &\text{ donné} \\ \frac{d\vec{z}^n}{dt} - \vec{G}(\vec{z}^n) + \delta \vec{z}^n - \frac{\partial \vec{G}}{\partial \vec{z}}(\vec{z}^n) \cdot d\vec{z} &= 0 \quad n=0,1,2,\dots \\ \vec{z}^{n+1} &= \vec{z}^n + d\vec{z} \end{aligned} \quad (1.7)$$

Mc Gill et Kenneth (1963) montrent alors le théorème suivant:

Si \vec{G} est continûment différentiable par rapport à \vec{z}

$\frac{\partial \vec{G}}{\partial \vec{z}}$ est lipschitzienne par rapport à \vec{z}

et \vec{z}^0 est dans un voisinage de la fonction linéaire $\vec{c}(t)$

$\vec{c}(t) = \frac{1}{t_2 - t_1} \cdot \left((\vec{z}_2 - \vec{z}_1) \cdot t + t_2 \vec{z}_1 - t_1 \vec{z}_2 \right)$ avec

$\max_{i,t} | \vec{z}^0(t) - \vec{c}(t) | \leq K$

alors pour $t_2 - t_1$ assez petit, la suite converge vers \vec{z}^* , unique solution de (2.6)

avec la propriété suivante:

il existe une constante k telle que

$$\max_{i,t} | \vec{z}^{k+1} - \vec{z}^* | < \frac{k}{(1-k)^2(nK+1)} \left(\max_{i,t} | \vec{z}^{k+1} - \vec{z}^k | \right)^2$$

Ce théorème montre la convergence quadratique de la suite de fonctions et permet de généraliser la méthode aux problèmes d'équations différentielles avec conditions aux deux bouts.

En remarquant que l'équation différentielle du second ordre peut se mettre sous la forme d'une équation différentielle du premier ordre moyennant l'augmentation de la dimension de \vec{z} , ce théorème peut s'appliquer aux systèmes du premier ordre avec conditions aux deux bouts du type:

$$\begin{aligned} \frac{d}{dt} \begin{pmatrix} \dot{\vec{z}} \\ \vec{z} \end{pmatrix} - \begin{pmatrix} \vec{G}(\vec{z}, t) \\ \dot{\vec{z}} \\ \vec{z} \end{pmatrix} &= 0 & (1.8) \\ (\mathbf{0}_n \ \mathbf{I}_n) \cdot \begin{pmatrix} \dot{\vec{z}} \\ \vec{z} \end{pmatrix} (t_1) &= \vec{z}_1 \\ (\mathbf{0}_n \ \mathbf{I}_n) \cdot \begin{pmatrix} \dot{\vec{z}} \\ \vec{z} \end{pmatrix} (t_2) &= \vec{z}_2 \end{aligned}$$

Or, les équations d'optimalité obtenues par le principe de Pontryagin aboutissent à un système différentiel avec conditions aux deux bouts qui peut parfois être mis sous cette forme. Ainsi Mc Gill et Kenneth calculent par cet algorithme la solution de plusieurs problèmes d'optimisation (maximisation de la distance en temps fixé pour un alunissage, transfert d'orbite en temps minimum). Long (1965) utilise cette méthode pour calculer une trajectoire d'orbite et constate la très bonne vitesse de convergence. Leondes & Paines montrent avec des hypothèses plus restrictives la convergence de la suite construite pour résoudre:

$$\begin{aligned} \vec{z} - \vec{G}(\vec{z}, \vec{u}, t) &= 0 & (1.9) \\ \vec{z}(t_1) &= \vec{z}_1 \\ \vec{z}(t_2) &= \vec{z}_2 \end{aligned}$$

où \vec{u} est solution d'une fonction implicite $\vec{Y}(\vec{z}, \vec{u}, t) = 0$ et peut se mettre sous la forme $\vec{u} = \vec{u}(\vec{z}, t)$. Ils généralisent la méthode aux systèmes du premier ordre.

1.3. Application à la commande optimale

Nous avons vu que de façon générale, on peut résoudre par la quasilinearisation des systèmes d'équations différentielles du premier ordre avec conditions aux deux bouts. Reprenons les équations d'optimalité (1.14), (1.15), (1.16), ces équations peuvent s'écrire:

$$\dot{\vec{x}} - \vec{f}(\vec{x}, \vec{u}, t) = 0 \quad (1.10)$$

$$\dot{\vec{\lambda}}^T + \frac{\partial H}{\partial \vec{x}}(\vec{x}, \vec{\lambda}, \vec{u}, t) = 0 \quad (1.11)$$

$$\frac{\partial H}{\partial \vec{u}}(\vec{x}, \vec{\lambda}, \vec{u}, t) = 0 \quad (1.12)$$

$$\vec{\lambda}^T(t_f) - \frac{\partial \Phi}{\partial \vec{x}}(\vec{x}_f, \vec{v}, t_f) = 0 \quad (1.13)$$

$$\vec{\Psi}(\vec{x}_f, t_f) = 0 \quad (1.14)$$

De façon formelle on peut donc appliquer la méthode définie précédemment pour calculer $\vec{x}(t)$, $\vec{\lambda}(t)$, $\vec{u}(t)$ et \vec{v} qui sera considéré ici comme une fonction constante. $\vec{u}(t)$ est exprimée en fonction de $\vec{x}(t)$ et de $\vec{\lambda}(t)$ par résolution de l'équation implicite $\partial H / \partial \vec{u} = 0$. Remarquons que les conditions finales ici sont plus complexes que celles considérées dans les théorèmes précédents. La suite des fonctions sera alors définie par :

$$\dot{\vec{x}}^n - \vec{f}(\vec{x}^n, \vec{u}^n, t) + d\vec{x} \cdot \frac{\partial \vec{f}}{\partial \vec{x}} d\vec{x} - \frac{\partial \vec{f}}{\partial \vec{u}} d\vec{u} = 0 \quad (1.15)$$

$$\dot{\vec{\lambda}}^n + \frac{\partial H^T}{\partial \vec{x}}(\vec{x}^n, \vec{\lambda}^n, \vec{u}^n, t) + d\vec{\lambda} + \frac{\partial^2 H}{\partial \vec{x}^2} d\vec{x} + \frac{\partial^2 H}{\partial \vec{x} \partial \vec{u}} d\vec{u} + \frac{\partial \vec{f}^T}{\partial \vec{x}} d\vec{\lambda} = 0 \quad (1.16)$$

$$\frac{\partial H}{\partial \vec{u}}(\vec{x}^n, \vec{\lambda}^n, \vec{u}^n, t) + \frac{\partial^2 H}{\partial \vec{u} \partial \vec{x}} d\vec{x} + \frac{\partial^2 H}{\partial \vec{u}^2} d\vec{u} + \frac{\partial \vec{f}^T}{\partial \vec{u}} d\vec{\lambda} = 0 \quad (1.17)$$

$$\vec{\lambda}^n(t_f) - \frac{\partial \Phi^T}{\partial \vec{x}}(\vec{x}_f, \vec{v}, t_f) + d\vec{\lambda}_f - \frac{\partial^2 \Phi}{\partial \vec{x}^2} d\vec{x}_f + \frac{\partial \vec{\Psi}^T}{\partial \vec{x}} d\vec{v} = 0 \quad (2.18)$$

$$\vec{\Psi}(\vec{x}_f, t_f) + \frac{\partial \vec{\Psi}}{\partial \vec{x}} d\vec{x}_f = 0 \quad (2.19)$$

La résolution de ce système linéaire nous permet d'obtenir les termes suivant de la suite de fonctions:

$$\begin{aligned}\vec{x}^{n+1} &= \vec{x}^n + d\vec{x} \\ \vec{\lambda}^{n+1} &= \vec{\lambda}^n + d\vec{\lambda} \\ \vec{u}^{n+1} &= \vec{u}^n + d\vec{u} \\ \vec{v}^{n+1} &= \vec{v}^n + d\vec{v}\end{aligned}$$

Pour être tout à fait rigoureux, il faudrait généraliser les théorèmes précédents à ce type de problèmes. Mitter (1966) par une autre démarche définit de façon rigoureuse la démarche de Newton. Cette démarche se rapproche de celle signalée dans le paragraphe 2.1. pour la recherche du minimum d'une fonction numérique à variables vectorielles. Cette approche est similaire aux méthodes du second ordre (Mitter 1966) détaillées au chapitre 1. La différence vient de la nature des nominales. Pour la méthode du second ordre, les nominales vérifient les conditions du premier ordre, ce qui n'est pas le cas dans les hypothèses de la méthode de quasilinéarisation.

traitement des contraintes:

Miele (1974) propose un algorithme basé sur la philosophie de la quasilinéarisation pour résoudre des problèmes de commande optimale soumis à des contraintes égalités du type

$$\vec{C}(\vec{x}, \vec{u}, t) = 0 \quad \left(\text{avec } \frac{\partial \vec{C}}{\partial \vec{u}} \neq 0 \right)$$

Pour traiter des contraintes, il définit le Hamiltonien augmenté $\bar{H} = H + \vec{\mu}^T \cdot \vec{C}$. Aux équations (2.15), (2.16), (2.17), (2.18) et (2.19) (où H est remplacé par le hamiltonien augmenté et où il faut tenir compte des variation de $\vec{\mu}$) s'ajoute une équation:

$$\vec{C}(\vec{x}^n, \vec{u}^n, t) + \frac{\partial \vec{C}}{\partial \vec{x}} d\vec{x} + \frac{\partial \vec{C}}{\partial \vec{u}} d\vec{u} = 0 \quad (1.20)$$

Remarquons que la minimisation de l'Hamiltonien n'est pas menée directement, elle sera assurée par

$$\frac{\partial \bar{H}}{\partial \vec{u}} = 0 \quad (1.21)$$

dans la mesure où $\frac{\partial^2 \bar{H}}{\partial \vec{u}^2} < 0$. Ce qu'il faudra vérifier en fin de convergence.

Nous nous proposons de développer une variante de la méthode de quasilinéarisation où nous intégrerons la minimisation de l'Hamiltonien. Cette nouvelle variante traitera des contraintes inégalité courantes dépendant de l'état et de la commande.

2. DEVELOPPEMENT D'UNE NOUVELLE VARIANTE

2.1. Présentation générale

Considérons le problème de la commande optimale avec instant finale fixé et en présence de contraintes inégalité du type:

$$C_i(\vec{u}, \vec{x}) \leq 0 \quad i = 1 \dots n_c \quad (2.1)$$

avec n_c contraintes $n_c \leq n_u$

et nous poserons l'hypothèse suivante:

$$\forall i \quad \frac{\partial C_i}{\partial u_i} \neq 0 \text{ et } \frac{\partial C_i}{\partial u_k} = 0 \quad \forall k \neq i \quad (2.2)$$

Pour plus de clarté, nous avons associé à chacune des contraintes le même indice que celui de la commande sur laquelle elle s'applique. Dans ce cas $\vec{\partial C} / \vec{\partial u}$ sera une matrice diagonale. Nous noterons J l'ensemble des indices correspondant aux contraintes saturées n'annulant pas $\partial C_i / \partial u_i$.

Considérons des fonctions état et adjoint nominales \vec{x} et $\vec{\lambda}$ données correspondant à une commande nominale donnée \vec{u} et qui ne vérifient pas nécessairement les conditions initiales et finales. Supposons qu'il existe une fonction $\vec{u}^*(\vec{x}, \vec{\lambda})$, différentiable par rapport à \vec{x} et $\vec{\lambda}$, telle que pour ces fonctions nominales (non optimales), le hamiltonien soit minimisé compte tenu des contraintes imposées. Dans ce cas, sauf si $\vec{u} = \vec{u}^*$ auquel cas le problème est déjà résolu, nous aurons:

$$\frac{\dot{\vec{x}}}{\vec{x}} - \vec{f}^* \left(\vec{x}(t), \vec{\lambda}(t), t \right) \neq 0 \quad (2.3)$$

$$\frac{\dot{\vec{\lambda}}}{\vec{\lambda}} - \vec{h}^* \left(\vec{x}(t), \vec{\lambda}(t), t \right) \neq 0 \quad (2.4)$$

où $\vec{f}^* \left(\vec{x}(t), \vec{\lambda}(t), t \right) = \vec{f} \left(\vec{x}(t), \vec{u}^* \left(\vec{x}, \vec{\lambda} \right), t \right)$

et
$$\vec{h}^* \left(\vec{x}(t), \vec{\lambda}(t), t \right) = - \frac{\partial H^T}{\partial \vec{x}} \left(\vec{x}(t), \vec{u}^* \left(\vec{x}, \vec{\lambda} \right), t \right) + \left(\frac{\partial H}{\partial \vec{u}} \left(\frac{\partial \vec{C}}{\partial \vec{u}} \right)^{-1} \cdot \frac{\partial C}{\partial \vec{x}} \right)^T \text{ restreint à } J$$

$$\vec{x}(t_0) - \vec{x}_0 \neq 0 \quad (2.5)$$

$$\vec{\Psi}(\vec{x}(t_f), t_f) \neq 0 \quad (2.6)$$

$$\vec{\lambda}(t_f) - \frac{\partial \Phi}{\partial \vec{x}}(\vec{x}(t_f), t_f, \vec{v}) \neq 0 \quad (2.7)$$

La commande qui minimise H sera définie par \vec{u}^* telle que

$$\begin{aligned} \text{pour } \vec{u}_{opt} &= (u_{opt_i}) = (\operatorname{argmin} H) \\ u_i^* &= u_{opt_i} && \text{si } C_i(\vec{x}, u_{opt_i}) \leq 0 \\ u_i^* &= \operatorname{Arg}(C_j(\vec{x}, u_j) = 0) && \text{sinon} \end{aligned}$$

A chaque itération, le domaine de saturation des contraintes sera déterminé afin de définir le système adjoint correspondant.

Nous nous proposons d'appliquer la méthode de Newton-Raphson généralisée pour résoudre ces équations. L'algorithme que nous allons développer consistera à calculer de façon itérative les fonctions \vec{x} , $\vec{\lambda}$ et le multiplicateur \vec{v} qui vérifieront ces équations. Une itération se fera en deux étapes. La première consistera à calculer les équations de variations, la seconde étape à résoudre le problème linéarisé. Nous proposerons une manière particulière de gérer les itérations et nous exposerons les possibilités d'initialiser l'algorithme.

Notations : Nous appellerons désormais "itération" l'ensemble des opérations nécessaires au calcul d'un terme de la suite. Ainsi l'itération N correspond au calcul du terme N de la suite. Pour plus de clarté, nous ne noterons pas les indices correspondant à une itération. Les variations d'une itération à l'autre seront notées $\delta\vec{x}$, $\delta\vec{\lambda}$ et $\delta\vec{v}$.

2.2. Description d'une itération

Comme nous l'avons signalé, une itération nécessitera deux étapes. La première étape consistera à calculer le système de variations, la seconde à résoudre le système ainsi obtenu. Nous verrons que cette dernière donnera lieu à plusieurs algorithmes possibles.

2.2.1. Calcul de variations

Appliquons le principe de la méthode de Newton-Raphson généralisé. Les variations des fonctions sont données par:

$$\left(\overset{\circ}{\vec{x}} - \vec{f}^* (\vec{x}(t), \vec{\lambda}(t), t) \right) + d \left(\overset{\circ}{\vec{x}} - \vec{f}^* (\vec{x}(t), \vec{\lambda}(t), t) \right) = 0 \quad (2.8)$$

$$\left(\overset{\circ}{\vec{\lambda}} - \vec{h}^* (\vec{x}(t), \vec{\lambda}(t), t) \right) + d \left(\overset{\circ}{\vec{\lambda}} - \vec{h}^* (\vec{x}(t), \vec{\lambda}(t), t) \right) = 0 \quad (2.9)$$

$$\left(\vec{x}(t_0) - \vec{x}_0 \right) + d \left(\vec{x}(t_0) - \vec{x}_0 \right) = 0 \quad (2.10)$$

$$\left(\vec{\psi}(\vec{x}(t_f), t_f) \right) + d \left(\vec{\psi}(\vec{x}(t_f), t_f) \right) = 0 \quad (2.11)$$

$$\left(\vec{\lambda}^T(t_f) - \partial \Phi / \partial \vec{x} (\vec{x}(t_f), t_f, \vec{v}) \right) + d \left(\vec{\lambda}^T(t_f) - \partial \Phi / \partial \vec{x} (\vec{x}(t_f), t_f, \vec{v}) \right) = 0 \quad (2.12)$$

Nous distinguerons les variations sur la dynamique et celles sur les contraintes.

Calculs de variations sur la dynamique

En supposant que les variations $\delta \vec{x}$ et $\delta \vec{\lambda}$ sont assez "petites" pour être assimilées à des variations du premier ordre, nous aurons :

$$\delta \vec{x} = d \vec{x}$$

$$\delta \vec{\lambda} = d \vec{\lambda}$$

$$d \left(\overset{\circ}{\vec{x}} - \vec{f}^* (\vec{x}(t), \vec{\lambda}(t), t) \right) = d \overset{\circ}{\vec{x}} - \frac{\partial \vec{f}^*}{\partial \vec{x}} \cdot d \vec{x} - \frac{\partial \vec{f}^*}{\partial \vec{\lambda}} \cdot d \vec{\lambda}$$

$$d \left(\overset{\circ}{\vec{\lambda}} - \vec{h}^* (\vec{x}(t), \vec{\lambda}(t), t) \right) = d \overset{\circ}{\vec{\lambda}} - \frac{\partial \vec{h}^*}{\partial \vec{x}} \cdot d \vec{x} - \frac{\partial \vec{h}^*}{\partial \vec{\lambda}} \cdot d \vec{\lambda}$$

ce qui donne :

$$\delta \overset{\circ}{\vec{x}} = \frac{\partial \vec{f}^*}{\partial \vec{x}} \cdot \delta \vec{x} + \frac{\partial \vec{f}^*}{\partial \vec{\lambda}} \cdot \delta \vec{\lambda} + \left(\vec{f}^* (\vec{x}, \vec{\lambda}) - \overset{\circ}{\vec{x}} \right) \quad (2.13)$$

$$\delta \overset{\circ}{\vec{\lambda}} = \frac{\partial \vec{h}^*}{\partial \vec{x}} \cdot \delta \vec{x} + \frac{\partial \vec{h}^*}{\partial \vec{\lambda}} \cdot \delta \vec{\lambda} + \left(\vec{h}^* (\vec{x}, \vec{\lambda}) - \overset{\circ}{\vec{\lambda}} \right) \quad (2.14)$$

Sur les intervalles où les contraintes ne sont pas saturées, nous aurons:

$$\frac{\partial \vec{f}^*}{\partial \vec{x}} = \frac{\partial \vec{f}}{\partial \vec{x}} - \frac{\partial \vec{f}}{\partial \vec{u}} \left(\frac{\partial^2 \mathbf{H}}{\partial \vec{u}^2} \right)^{-1} \cdot \frac{\partial^2 \mathbf{H}}{\partial \vec{u} \partial \vec{x}} \quad (2.15)$$

$$\frac{\partial \vec{f}^*}{\partial \vec{\lambda}} = - \frac{\partial \vec{f}}{\partial \vec{u}} \left(\frac{\partial^2 \mathbf{H}}{\partial \vec{u}^2} \right)^{-1} \cdot \frac{\partial \vec{f}^T}{\partial \vec{u}} \quad (2.16)$$

$$\frac{\partial \vec{h}^*}{\partial \vec{x}} = \frac{\partial^2 \mathbf{H}}{\partial \vec{x}^2} - \frac{\partial^2 \mathbf{H}}{\partial \vec{x} \partial \vec{u}} \left(\frac{\partial^2 \mathbf{H}}{\partial \vec{u}^2} \right)^{-1} \cdot \frac{\partial^2 \mathbf{H}}{\partial \vec{u} \partial \vec{x}} \quad (2.1)$$

et

$$\frac{\partial \vec{h}^*}{\partial \vec{\lambda}} = - \frac{\partial \vec{f}^{*T}}{\partial \vec{x}} \quad (2.18)$$

Sur les intervalles où les contraintes sont saturées:

$$\frac{\partial \vec{f}^*}{\partial \vec{x}} = \frac{\partial \vec{f}}{\partial \vec{x}} - \frac{\partial \vec{f}}{\partial \vec{u}} \left(\frac{\partial \vec{C}}{\partial \vec{u}} \right)^{-1} \cdot \frac{\partial \vec{C}}{\partial \vec{x}} \quad (2.19)$$

$$\frac{\partial \vec{f}^*}{\partial \vec{\lambda}} = 0 \quad (2.20)$$

$$\begin{aligned} \frac{\partial \vec{h}^*}{\partial \vec{x}} = & - \frac{\partial^2 \mathbf{H}}{\partial \vec{x}^2} + \frac{\partial^2 \mathbf{H}}{\partial \vec{x} \partial \vec{u}} \left(\frac{\partial \vec{C}}{\partial \vec{u}} \right)^{-1} \cdot \frac{\partial \vec{C}}{\partial \vec{x}} + \frac{\partial \vec{C}^T}{\partial \vec{x}} \left(\frac{\partial \vec{C}}{\partial \vec{u}} \right)^{-1} \frac{\partial \mathbf{H}}{\partial \vec{u} \partial \vec{x}} \\ & - \frac{\partial \vec{C}^T}{\partial \vec{x}} \left(\frac{\partial \vec{C}}{\partial \vec{u}} \right)^{-1} \left(\frac{\partial^2 \mathbf{H}}{\partial \vec{u}^2} \right) \left(\frac{\partial \vec{C}}{\partial \vec{u}} \right)^{-1} \cdot \frac{\partial \vec{C}}{\partial \vec{x}} \end{aligned} \quad (2.21)$$

et

$$\frac{\partial \vec{h}^*}{\partial \vec{\lambda}} = - \frac{\partial \vec{f}^{*T}}{\partial \vec{x}} \quad (2.22)$$

Calculs de variations sur les conditions initiales et finales

On calcule, de la même façon que pour la dynamique, les variations au premier ordre sur les condition initiales et finales.

A l'instant initial, nous avons :

$$d \left(\vec{x}(t_0) - \vec{x}_0 \right) = d\vec{x}(t_0) \quad (2.23)$$

A l'instant final, nous avons :

$$d(\vec{\psi}) = \frac{\partial \vec{\psi}}{\partial \vec{x}} \cdot d\vec{x}(t_f) \quad (2.24)$$

$$d\left(\vec{\lambda}(t_f) - \frac{\partial \Phi^T}{\partial \vec{x}}\right) = d\vec{\lambda}(t_f) - \frac{\partial^2 \Phi}{\partial \vec{x}^2} \cdot d\vec{x}(t_f) - \frac{\partial \vec{\psi}^T}{\partial \vec{x}} \cdot d\vec{v} \quad (2.25)$$

Ce qui donne les équations suivantes :

$$\delta \vec{x}(t_0) + \left(\vec{x}(t_0) - \vec{x}_0 \right) = 0 \quad (2.26)$$

$$\frac{\partial \vec{\psi}}{\partial \vec{x}} \delta \vec{x}(t_f) + \left(\vec{\psi} \right) = 0 \quad (2.27)$$

$$\delta \vec{\lambda}(t_f) - \frac{\partial^2 \Phi}{\partial \vec{x}^2} \cdot \delta \vec{x}(t_f) - \frac{\partial \vec{\psi}^T}{\partial \vec{x}} \cdot \delta \vec{v} + \left(\vec{\lambda}(t_f) - \frac{\partial \Phi^T}{\partial \vec{x}} \right) = 0 \quad (2.28)$$

Il s'agit maintenant de résoudre le système d'équations différentielles linéaires formé par les équations (2.13), (2.14), (2.26), (2.27) et (2.28).

2.2.2. Résolution du système différentiel linéarisé

Les méthodes utilisées dans la littérature pour résoudre ce système sont nombreuses (Dyer, Mc Reynolds 1970, Keller 1976). La plus classique de ces méthodes utilise la matrice de transition. La solution obtenue dépendra de cette matrice et de paramètres déterminés à partir des conditions finales. Cette matrice pourra être calculée soit par intégration d'une équation différentielle matricielle soit par perturbation des conditions initiales. Cette dernière approche correspond au calcul direct des sensibilités des conditions finales par rapport aux conditions initiales. Nous signalerons enfin une méthode qui s'inspire de la résolution de Kallman pour les problèmes linéaires quadratiques. Celle-ci aboutit aux équations différentielles matricielles de RICCATI.

Résolution par la matrice de transition.

Considérons \mathbf{T} , la matrice de transition directe du système linéarisé définie par:

$$\dot{\mathbf{T}} = \begin{pmatrix} \frac{\partial \vec{f}^*}{\partial \vec{x}} & \frac{\partial \vec{f}^*}{\partial \vec{\lambda}} \\ \frac{\partial \vec{h}^*}{\partial \vec{x}} & \frac{\partial \vec{h}^*}{\partial \vec{\lambda}} \end{pmatrix} \cdot \mathbf{T} \quad (2.29)$$

$$\mathbf{T}(t_0) = \mathbf{I}_{2n_x} \text{ (matrice identité de } \mathbb{R}^{2n_x}\text{)}$$

alors la solution est donnée par

$$\begin{pmatrix} \delta \vec{x}(t) \\ \delta \vec{\lambda}(t) \end{pmatrix} = \mathbf{T}(t) \cdot \begin{pmatrix} \vec{a} \\ \vec{b} \end{pmatrix} + \begin{pmatrix} \delta \vec{x}^P(t) \\ \delta \vec{\lambda}^P(t) \end{pmatrix} \quad (2.30)$$

$\delta \vec{x}^P(t)$ et $\delta \vec{\lambda}^P(t)$ sont les solutions du système complet pour des conditions initiales données que nous précisons plus loin. \vec{a} et \vec{b} sont des paramètres qui seront déterminés par les conditions initiales (2.26) d'une part et finales (2.27), (2.28) d'autre part. En posant:

$$\delta \vec{x}^P(t_0) = \vec{x}_0 - \vec{x}(t_0) \quad (2.31)$$

$$\delta \vec{\lambda}^P(t_0) = 0 \quad (2.32)$$

et en tenant compte de l'équation (2.47), nous obtenons directement

$$\vec{a} = 0 \quad (2.33)$$

$$\vec{b} = \delta \vec{\lambda}(t_0) \quad (2.34)$$

Ce choix de conditions initiales pour la solution particulière permet de simplifier les calculs. En effet, si nous adoptons la notation par blocs :

$$\mathbf{T}(t) = \begin{pmatrix} \mathbf{T}_{11}(t) & \mathbf{T}_{12}(t) \\ \mathbf{T}_{21}(t) & \mathbf{T}_{22}(t) \end{pmatrix} \quad (2.35)$$

nous remarquons que seules les sous matrices \mathbf{T}_{12} et \mathbf{T}_{22} sont nécessaires au calcul de la solution. Nous ne calculerons donc que la "moitié" de la matrice de transition.

A l'instant final nous avons alors:

$$\begin{pmatrix} \delta \vec{x}(t_f) \\ \delta \vec{\lambda}(t_f) \end{pmatrix} = \mathbf{T}(t_f) \cdot \begin{pmatrix} 0 \\ \delta \vec{\lambda}(t_0) \end{pmatrix} + \begin{pmatrix} \delta \vec{x}^P(t_f) \\ \delta \vec{\lambda}^P(t_f) \end{pmatrix} \quad (2.36)$$

En tenant compte des équations (2.27), (2.28), nous avons:

$$\mathbf{W} \cdot \begin{pmatrix} \delta \vec{\lambda}(t_0) \\ \delta \vec{v} \end{pmatrix} = \vec{w} \quad (2.37)$$

avec

$$\mathbf{W} = \begin{pmatrix} \frac{\partial^2 \Phi}{\partial \vec{x}^2} \mathbf{T}_{12}(t_f) - \mathbf{T}_{22}(t_f) & \frac{\partial \vec{\psi}^T}{\partial \vec{x}} \\ \frac{\partial \vec{\psi}}{\partial \vec{x}} \cdot \mathbf{T}_{12}(t_f) & 0 \end{pmatrix} \quad (2.38)$$

et

$$\vec{w} = \begin{pmatrix} \delta \vec{\lambda}^P(t_f) - \frac{\partial^2 \Phi}{\partial \vec{x}^2} \cdot \delta \vec{x}^P(t_f) + (\vec{\lambda}(t_f) - \frac{\partial \Phi^T}{\partial \vec{x}}) \\ - \frac{\partial \vec{\psi}}{\partial \vec{x}} \cdot \delta \vec{x}^P(t_f) - \vec{\psi} \end{pmatrix} \quad (2.39)$$

On retrouve parfois dans la littérature, cette méthode sous le nom de "méthode de superposition". En effet, calculer \mathbf{T}_{12} et \mathbf{T}_{22} revient à calculer les n_x solutions du système homogène correspondant aux n_x conditions initiales suivantes :

$$\begin{aligned} \delta \vec{x}^i(t_0) &= 0 \\ \delta \vec{\lambda}^i(t_0) &= (0, \dots, 0, 1, 0, \dots, 0)^T \end{aligned}$$

↑
ième composante

La solution générale du système sera alors donnée par la superposition de ces solutions et de la solution du système complet pour les conditions initiales définies précédemment.

$$\delta \vec{x}(t) = \delta \vec{x}^p(t) + \sum_{i=1}^{nx} \delta \lambda_{oi} \cdot \delta \vec{x}^i(t) \quad (2.40)$$

$$\delta \vec{\lambda}(t) = \delta \vec{\lambda}^p(t) + \sum_{i=1}^{nx} \delta \lambda_{oi} \cdot \delta \vec{\lambda}^i(t) \quad (2.41)$$

Résolution par la méthode des sensibilités directes

Remarquons que les matrices $T_{12}(t_f)$ et $T_{22}(t_f)$ ne sont autres que les matrices

$$\frac{\partial \vec{x}_f}{\partial \vec{\lambda}_o} \quad \text{et} \quad \frac{\partial \vec{\lambda}_f}{\partial \vec{\lambda}_o}$$

qui sont les sensibilités des valeurs finales $\vec{x}(t_f)$ et $\vec{\lambda}(t_f)$ par rapport à une variation de la valeur initiale de l'adjoint. En utilisant cette interprétation, nous pouvons calculer des valeurs approchées de $T_{12}(t_f)$ et $T_{22}(t_f)$ par calcul de sensibilités.

Considérons des valeurs initiales nominales \vec{x}_o et $\vec{\lambda}_o$ et une variation $\delta \vec{\lambda}_o^i$ donnée par:

$$\delta \vec{\lambda}_o^i = (0, \dots, 0, \delta \lambda_{oi}, 0, \dots, 0)^T$$

Considérons les valeurs finales \vec{x}_f et $\vec{\lambda}_f$ obtenues à partir des valeurs nominales et celles obtenues par intégration à partir des valeurs \vec{x}_o et $\vec{\lambda}_o + \delta \vec{\lambda}_o^i$, du système différentiel non linéaire formé par les équation d'état et les équations adjointes. Nous noterons ces dernières $\vec{x}_f + \delta \vec{x}_f^i$ et $\vec{\lambda}_f + \delta \vec{\lambda}_f^i$. Alors, si $\delta \vec{x}_f$ et $\delta \vec{\lambda}_f$ sont petits, nous aurons :

$$\frac{\partial \vec{x}_f}{\partial \lambda_{oi}} \approx \frac{\delta \vec{x}_f^i}{\delta \lambda_{oi}} \quad (2.42)$$

$$\frac{\partial \vec{\lambda}_f}{\partial \lambda_{oi}} \approx \frac{\delta \vec{\lambda}_f^i}{\delta \lambda_{oi}} \quad (2.43)$$

Ainsi nous pourrions calculer $T_{12}(t_f)$ et $T_{22}(t_f)$ par $nx+1$ intégrations directes du système non linéaire formé par l'état et l'adjoint à condition que les valeurs de $\delta \vec{x}_f$ et $\delta \vec{\lambda}_f$ soient petits.

L'avantage de cette méthode réside dans la diminution du nombre d'opérations numériques pour le calcul des matrices $T_{12}(t_f)$ et $T_{22}(t_f)$. En effet pour la méthode

précédente, il était nécessaire de calculer à chaque instant la matrice du système linéarisé à partir des fonctions état et adjoint de l'itération précédente, ce qui implique un temps de calcul important pour l'itération. Pour la méthode des sensibilités, il suffit de $n_x + 1$ intégrations directes pour obtenir $\mathbf{T}_{12}(t_f)$ et $\mathbf{T}_{22}(t_f)$.

Résolution par les équations de RICCATI

Cette méthode de résolution s'inspire de la méthode développée par Mitter pour les variations secondes. L'intérêt de cette méthode réside dans le fait que les intégrations d'équations différentielles nécessaires au calcul des variations $\delta\vec{x}$, $\delta\vec{\lambda}$ et $\delta\vec{v}$ sont stables. Nous présentons tout d'abord cette approche appliquée au contexte de la quasilinearisation, nous discuterons ensuite des limites de cette approche.

$$\text{Si nous posons } \begin{pmatrix} \delta\vec{\lambda}(t) \\ 0 \end{pmatrix} = \begin{pmatrix} \mathbf{R}(t) & \mathbf{S}(t)^T \\ \mathbf{S}(t) & \mathbf{Q}(t) \end{pmatrix} \cdot \begin{pmatrix} \delta\vec{x}(t) \\ \delta\vec{v} \end{pmatrix} + \begin{pmatrix} \vec{\mu} \\ \vec{\sigma} \end{pmatrix} \quad (2.44)$$

les équations (2.27) et (2.28) imposent

$$\mathbf{R}(t_f) = \frac{\partial^2 \Phi}{\partial \vec{x}^2} \quad (2.45)$$

$$\mathbf{S}(t_f) = \frac{\partial \vec{\psi}}{\partial \vec{x}} \quad (2.46)$$

$$\mathbf{Q}(t_f) = 0 \quad (2.47)$$

$$\vec{\mu}(t_f) = - \left(\vec{\lambda}_f - \frac{\partial \Phi^T}{\partial \vec{x}} \right) \quad (2.48)$$

$$\vec{\sigma}(t_f) = \vec{\psi} \quad (2.49)$$

En dérivant (2.44) et en remplaçant $\delta\vec{x}$ et $\delta\vec{\lambda}$ par leurs expressions, nous obtenons des équations différentielles matricielles qui déterminent \mathbf{R} , \mathbf{S} , \mathbf{Q} , $\vec{\mu}$ et $\vec{\sigma}$:

$$\dot{\mathbf{R}} + \mathbf{R} \cdot \frac{\partial \vec{f}^*}{\partial \vec{x}} + \frac{\partial \vec{f}^{*T}}{\partial \vec{x}} \cdot \mathbf{R} + \mathbf{R} \cdot \frac{\partial \vec{f}^*}{\partial \vec{\lambda}} \cdot \mathbf{R} - \frac{\partial \vec{h}^*}{\partial \vec{x}} = 0 \quad (2.50)$$

$$\dot{\mathbf{S}} + \mathbf{S} \cdot \left(\frac{\partial \vec{f}^*}{\partial \vec{x}} + \frac{\partial \vec{f}^{*T}}{\partial \vec{\lambda}} \mathbf{R} \right) = 0 \quad (2.51)$$

$$\dot{\mathbf{Q}} + \mathbf{S} \cdot \frac{\partial \vec{f}^*}{\partial \vec{\lambda}} \cdot \mathbf{S}^T = 0 \quad (2.52)$$

$$\vec{\mu} + \left(\frac{\partial \vec{f}^{*T}}{\partial \vec{x}} + \mathbf{R} \cdot \frac{\partial \vec{f}^*}{\partial \vec{\lambda}} \right) \cdot \vec{\mu} + \mathbf{R} \cdot \left(\vec{f}^* - \vec{x} \right) - \left(\vec{h}^* - \vec{\lambda} \right) = 0 \quad (2.53)$$

$$\vec{\sigma} + \mathbf{S} \cdot \left(\frac{\partial \vec{f}^*}{\partial \vec{\lambda}} \cdot \vec{\mu} + \left(\vec{f}^* - \vec{x} \right) \right) = 0 \quad (2.54)$$

L'intégration en sens inverse de ces équations permet alors d'obtenir la variation $\delta \vec{v}$ directement en fonction de la variation des contraintes finales $\Delta \vec{\psi} = -\vec{\psi}$ et de la variation des conditions initiales $\delta \vec{x}_o$

$$\delta \vec{v} = \mathbf{Q}(t_o)^{-1} \left(\mathbf{S}(t_o) \cdot \delta \vec{x}_o - \vec{\psi} \right) \quad (2.55)$$

Condition de validité de la méthode :

Pour valider cette méthode, il faudra s'assurer de l'existence d'une solution \mathbf{R} définie positive à l'équation (2.72). Cette existence sera assurée sous les conditions supplémentaires :

$$\begin{aligned} \frac{\partial^2 \Phi}{\partial \vec{x}^2} & \text{ définie positive} \\ \frac{\partial \vec{h}^*}{\partial \vec{x}} & \text{ semi définie positive} \end{aligned}$$

(la condition $\partial^2 H / \partial u^2$ définie positive est contenue implicitement dans le fait de calculer de la commande par minimisation de H). Ces conditions seront en général difficile à vérifier sauf lorsque les fonctions se trouvent dans le voisinage d'un minimum local.

Ce type de résolution sera donc très limité dans le cadre de notre méthode de quasilinearisation. Sauf dans les cas où nous serons sûrs de l'existence de \mathbf{R} , la résolution par matrices de Riccati ne sera pas envisageable. Nous signalerons à ce propos la méthode de Talwar et Sivan (1989) pour certains cas particuliers. Ces cas



particuliers correspondent aux systèmes linéaires quadratiques par rapport à la commande et non linéaires par rapport à l'état définis par les équations:

$$\dot{\vec{x}} = \vec{f}(\vec{x}(t), t) + \mathbf{B} \cdot \vec{u} \quad (2.56)$$

$$J = \int_{t_0}^{t_f} \left(L(\vec{x}(t), t) + \frac{1}{2} \cdot \vec{u}^T \cdot \mathbf{R} \cdot \vec{u} \right) dt + \varphi(\vec{x}_{f, t_f}) \quad (2.57)$$

Talwar et al. ont isolé dans ce cas, le facteur de divergence des équations de Riccati et proposent une méthode dérivée de la méthode de Newton-Raphson, avec résolution par les équations différentielles de Riccati.

2.2.3. Domaine de validité et généralisation

Validité de la méthode:

Comme nous l'avons indiqué au début du développement de la méthode, une des conditions imposées à l'application de cette méthode est la minimisation du Hamiltonien. Cette méthode s'applique donc lorsqu'il est possible de calculer simplement la commande qui minimise le Hamiltonien, en particulier lorsque nous disposons d'une expression analytique de cette dernière. En général, u^* sera obtenue en résolvant dans le domaine d'admissibilité de la commande l'équation implicite:

$$\frac{\partial H}{\partial \vec{u}} = 0$$

L'existence d'une commande optimale différentiable implique dans le cas où il n'existe pas de contraintes courantes, l'inversibilité de $\frac{\partial^2 H}{\partial \vec{u}^2}$.

Dans le cas de contraintes courantes égalité, en particulier lors de saturation de contraintes inégalité, le calcul de ce minimum revient à résoudre par rapport à \vec{u} et à $\vec{\mu}$ le système d'équations implicites :

$$\begin{aligned} \frac{\partial \bar{H}}{\partial \vec{u}} &= 0 \\ \vec{C} &= 0 \end{aligned}$$

L'hypothèse d'inversibilité précédente est alors remplacée par l'hypothèse d'inversibilité de $\frac{\partial^2 H}{\partial \vec{u}^2}$ et de $\frac{\partial \vec{C}}{\partial \vec{u}}$.



Bien que le traitement direct des contraintes d'inégalité que nous avons développé s'applique à un cas un peu particulier, la transformation en contraintes intégrales signalée au premier chapitre nous permet d'envisager l'application de notre méthode à des problèmes soumis à des contraintes courantes de forme très générales.

Problèmes à temps final libre:

Nous avons étudié, jusqu'à présent, le problème à temps terminal fixé. Pour le cas où t_f est libre, en particulier pour les problèmes de temps minimum, le temps devient un élément à optimiser au même titre que les commandes. Nous avons une condition d'optimalité supplémentaire donnée par la condition de transversalité:

$$\left(H(\vec{x}, \vec{u}^*, \vec{\lambda}, t) + \frac{\partial \Phi}{\partial t}(\vec{x}, t, \vec{v}) \right)_{t_f} = 0 \quad (2.58)$$

Le calcul de variation devra tenir compte de cette condition et les équations de variation sont de plus modifiées par la variation de l'instant final. Nous aurons alors:

$$d\vec{x}_f = \delta\vec{x}_f + \dot{\vec{x}}(t_f) \cdot \delta t_f \quad (2.59)$$

$$d\vec{\lambda}_f = \delta\vec{\lambda}_f + \dot{\vec{\lambda}}(t_f) \cdot \delta t_f \quad (2.60)$$

$$\frac{\partial \vec{\psi}}{\partial \vec{x}} \delta\vec{x}(t_f) + \frac{d\vec{\psi}}{dt} \cdot \delta t_f + \vec{\psi} = 0 \quad (2.61)$$

$$\begin{aligned} & \frac{\partial^2 \Phi}{\partial \vec{x}^2} \cdot \delta\vec{x}(t_f) + \frac{\partial \vec{\psi}^T}{\partial \vec{x}} \cdot d\vec{v} + \left(\frac{\partial}{\partial t} \left(\frac{\partial \Phi}{\partial \vec{x}} \right) + \frac{\partial^2 \Phi}{\partial \vec{x}^2} \cdot \dot{\vec{x}}(t_f) - \dot{\vec{\lambda}}(t_f) \right) \cdot \delta t_f \\ & + \delta\vec{\lambda}^T(t_f) - \left(\vec{\lambda}(t_f) - \frac{\partial \Phi^T}{\partial \vec{x}} \right) = 0 \end{aligned} \quad (2.62)$$

$$\begin{aligned} & \left(\frac{\partial H}{\partial \vec{x}} + \frac{\partial}{\partial t} \left(\frac{\partial \Phi}{\partial \vec{x}} \right) + \vec{f}^T \cdot \frac{\partial^2 \Phi}{\partial \vec{x}^2} \right) \cdot \delta\vec{x}(t_f) + \left(\frac{\partial \vec{\psi}}{\partial t} + \frac{\partial \vec{\psi}}{\partial \vec{x}} \cdot \vec{f} \right) \cdot d\vec{v} \\ & + \left(\vec{f}^T \cdot \left(\frac{\partial^2 \Phi^T}{\partial \vec{x} \partial t} + \frac{\partial^2 \Phi}{\partial \vec{x}^2} \cdot \dot{\vec{x}} \right) + \left(\frac{\partial H}{\partial \vec{x}} + \frac{\partial^2 \Phi}{\partial \vec{x} \partial t} \right) \cdot \dot{\vec{x}}_f + \frac{\partial^2 \Phi}{\partial t^2} \right) \cdot dt_f \\ & + \left(H + \frac{\partial \Phi}{\partial t} \right) - \vec{f}^T \cdot \left(\vec{\lambda}_f - \frac{\partial \Phi^T}{\partial \vec{x}} \right) = 0 \end{aligned} \quad (2.63)$$

Le système linéaire (2.37) devient ici

$$\begin{pmatrix} \frac{\partial^2 \Phi}{\partial \vec{x}^2} \cdot \mathbf{T}_{12}(t_f) - \mathbf{T}_{22}(t_f) & \frac{\partial \vec{\psi}^T}{\partial \vec{x}} & \vec{m} \\ \frac{\partial \vec{\psi}}{\partial \vec{x}} \cdot \mathbf{T}_{12}(t_f) & 0 & \vec{n} \\ \vec{m} & \vec{n} & \alpha \end{pmatrix} \cdot \begin{pmatrix} \delta \vec{\lambda}_o \\ \delta \vec{v} \\ \delta t_f \end{pmatrix} = \begin{pmatrix} \delta \vec{\lambda}^P(t_f) - \frac{\partial^2 \Phi}{\partial \vec{x}^2} \cdot \delta \vec{x}^P(t_f) + (\vec{\lambda}^T(t_f) - \frac{\partial \Phi^T}{\partial \vec{x}}) \\ - \frac{\partial \vec{\psi}}{\partial \vec{x}} \cdot \delta \vec{x}^P(t_f) - \vec{\psi} \\ \beta \end{pmatrix} \quad (2.64)$$

où
$$\vec{m} = \frac{d}{dt} \left(\frac{\partial \Phi}{\partial \vec{x}} \right)^T - \vec{\lambda}_f \quad (2.65)$$

$$\vec{n} = \frac{d\vec{\psi}}{dt} \quad (2.66)$$

$$\vec{m} = \frac{\partial H^T}{\partial \vec{x}} + \frac{\partial^2 \Phi^T}{\partial t \partial \vec{x}} + \frac{\partial^2 \Phi}{\partial \vec{x}^2} \cdot \vec{f} \quad (2.67)$$

$$\vec{n} = \frac{\partial \vec{\psi}}{\partial t} + \frac{\partial \vec{\psi}}{\partial \vec{x}} \cdot \vec{f} \quad (2.68)$$

$$\alpha = \left(\vec{f}^T \cdot \left(\frac{\partial^2 \Phi^T}{\partial \vec{x} \partial t} + \frac{\partial^2 \Phi}{\partial \vec{x}^2} \cdot \vec{x} \right) + \left(\frac{\partial H}{\partial \vec{x}} + \frac{\partial^2 \Phi}{\partial \vec{x} \partial t} \right) \cdot \vec{x}_f + \frac{\partial^2 \Phi}{\partial t^2} \right) \quad (2.69)$$

$$\beta = \left(H + \frac{\partial \Phi}{\partial t} \right) - \left(\vec{\lambda}_f^T - \frac{\partial \Phi}{\partial \vec{x}} \right) \cdot \vec{f} \quad (2.70)$$

Cas particulier :

Si on suppose que les dynamiques sont vérifiées et que seules les conditions terminales ne sont pas encore réalisées, on aura (cf annexe 1):

$$\dot{\vec{x}} = \vec{f}^*(\vec{x}, \vec{\lambda}) \quad (2.71)$$

$$\dot{\vec{\lambda}} = \vec{h}^*(\vec{x}, \vec{\lambda}) \quad (2.72)$$

$$\vec{m} = m \quad (2.73)$$

$$\vec{n} = n \quad (2.74)$$

$$\alpha = \frac{\partial}{\partial t} \left(H + \frac{d\Phi}{dt} \right) \quad (2.75)$$

et la matrice (2.64) sera symétrique.



Contraintes intégrales:

Dans le cas où il y a des contraintes intégrales du type

$$\xi(\vec{x}(t_f), t_f) + \int_{t_0}^{t_f} M(\vec{x}(t), \vec{u}(t), t) dt = 0$$

Nous définissons une nouvelle variable d'état a telle que

$$\begin{aligned} \dot{a} &= M(\vec{x}(t), \vec{u}(t), t) \\ a(t_0) &= 0 \\ a(t_f) + \xi(\vec{x}(t_f), t_f) &= 0 \end{aligned}$$

le Hamiltonien et les conditions finales sur l'adjoint deviennent

$$\begin{aligned} H &= L(\vec{x}(t), \vec{u}(t), t) + \vec{\lambda}^T \cdot \vec{f}(\vec{x}(t), \vec{u}(t), t) + \mu \cdot M(\vec{x}(t), \vec{u}(t), t) \\ \vec{\lambda}(t_f) &= \frac{\partial \Phi^T}{\partial \vec{x}} \quad \text{avec } \Phi = \varphi + \vec{v}^T \cdot \vec{\psi} + \mu \cdot \xi \end{aligned}$$

On a alors u* fonction de $\vec{x}, \vec{\lambda}, \mu$, et t et les équation de variations sont:

$$\delta \dot{\vec{x}} = \frac{\partial \vec{f}^*}{\partial \vec{x}} \cdot \delta \vec{x} + \frac{\partial \vec{f}^*}{\partial \vec{\lambda}} \cdot \delta \vec{\lambda} + \frac{\partial \vec{f}^*}{\partial \mu} \cdot \delta \mu - (\dot{\vec{x}} - \vec{f}^*) \quad (2.76)$$

$$\delta \dot{\vec{\lambda}} = \frac{\partial \vec{h}^*}{\partial \vec{x}} \cdot \delta \vec{x} + \frac{\partial \vec{h}^*}{\partial \vec{\lambda}} \cdot \delta \vec{\lambda} + \frac{\partial \vec{h}^*}{\partial \mu} \cdot \delta \mu - (\dot{\vec{\lambda}} - \vec{h}^*) \quad (2.77)$$

$$\delta \dot{a} = \frac{\partial M^*}{\partial \vec{x}} \cdot \delta \vec{x} + \frac{\partial M^*}{\partial \vec{\lambda}} \cdot \delta \vec{\lambda} + \frac{\partial M^*}{\partial \mu} \cdot \delta \mu \quad (2.78)$$

$$\delta \dot{\mu} = 0 \quad (2.79)$$

$$\vec{\psi}(\vec{x}(t_f), t_f) + d \vec{\psi}(\vec{x}(t_f), t_f) = 0 \quad (2.80)$$

$$\left(\vec{\lambda}(t_f) - \frac{\partial \Phi^T}{\partial \vec{x}} \right) + d \left(\vec{\lambda}(t_f) - \frac{\partial \Phi^T}{\partial \vec{x}} \right) = 0 \quad (2.81)$$

$$\left(H + \frac{\partial \Phi}{\partial t} \right) + d \left(H + \frac{\partial \Phi}{\partial t} \right) = 0 \quad (2.82)$$

$$\left(a(t_f) + \xi(\vec{x}(t_f), t_f) \right) + d \left(a(t_f) + \xi(\vec{x}(t_f), t_f) \right) = 0 \quad (2.83)$$

Au premier ordre on trouve:

$$\vec{\psi} + \frac{\partial \vec{\psi}}{\partial \vec{x}} \cdot d\vec{x}_f + \frac{\partial \vec{\psi}}{\partial t} \cdot dt_f = 0 \quad (2.84)$$

$$\left(\vec{\lambda}_f - \frac{\partial \Phi^T}{\partial \vec{x}} \right) + d\vec{\lambda}_f - \frac{\partial^2 \Phi}{\partial \vec{x}^2} \cdot d\vec{x}_f - \frac{\partial \vec{\psi}^T}{\partial \vec{x}} \cdot d\vec{v} - \frac{\partial \xi^T}{\partial \vec{x}} \cdot d\mu - \frac{\partial^2 \Phi^T}{\partial t \partial \vec{x}} \cdot dt_f = 0 \quad (2.85)$$

$$\begin{aligned} \left(H + \frac{\partial \Phi}{\partial t} \right) + \left(\frac{\partial H}{\partial \vec{x}} + \frac{\partial^2 \Phi}{\partial t \partial \vec{x}} \right) \cdot d\vec{x}_f + \vec{\Gamma}^T \cdot d\vec{\lambda}_f + \left(M^T + \frac{\partial \xi^T}{\partial t} \right) \cdot d\mu \\ + \frac{\partial \vec{\psi}^T}{\partial t} \cdot d\vec{v} + \left(\frac{\partial H}{\partial t} + \frac{\partial^2 \Phi}{\partial t^2} \right) \cdot dt_f = 0 \end{aligned} \quad (2.86)$$

$$(a + \xi) + \frac{\partial \xi}{\partial \vec{x}} \cdot d\vec{x}_f + da_f + \frac{\partial \xi}{\partial t} \cdot dt_f = 0 \quad (2.87)$$

Soit à l'instant final:

$$\begin{pmatrix} \delta \vec{\lambda} \\ 0 \\ -\delta a \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{\partial^2 \Phi}{\partial \vec{x}^2} & \frac{\partial \vec{\psi}^T}{\partial \vec{x}} & \frac{\partial \xi^T}{\partial \vec{x}} & \frac{d}{dt} \left(\frac{\partial \Phi}{\partial \vec{x}} \right) - \frac{\vec{\lambda}}{\lambda} \\ \frac{\partial \vec{\psi}}{\partial \vec{x}} & 0 & 0 & \frac{d\vec{\psi}}{dt} \\ \frac{\partial \xi}{\partial \vec{x}} & 0 & 0 & a + \frac{d\xi}{dt} \\ \vec{m}^T & \vec{n}_1^T & \vec{n}_2^T & \alpha \end{pmatrix} \cdot \begin{pmatrix} \delta \vec{x} \\ d\vec{v} \\ d\mu \\ dt \end{pmatrix} + \begin{pmatrix} -\left(\vec{\lambda}_f - \frac{\partial \Phi^T}{\partial \vec{x}} \right) \\ \vec{\psi} \\ (a + \xi) \\ \beta \end{pmatrix} \quad (2.88)$$

où $\vec{n}_2 = a + \left(\frac{\partial \xi}{\partial t} + \frac{\partial \xi}{\partial \vec{x}} \cdot \vec{\Gamma} \right)$

\vec{n}_1 est donné par (2.68)

\vec{m} par (2.67)

α par (2.69)

β par (2.70)

2.3. Gestion des itérations

2.3.1. Etude de la convergence

Si les premières applications de la méthode de quasilinearisation (Long 1965, Tapley et al. 1967, Mc Cue 1967) font apparaître une grande vitesse de convergence, elles révèlent aussi la grande sensibilité de la méthode à l'initialisation: le domaine de convergence de la méthode est très faible. Il est donc très important de trouver une initialisation assez proche de l'optimum ce qui se révèle relativement délicat. Plusieurs solutions peuvent être envisagées dans le but d'augmenter ce domaine.

Dans une première étude théorique menée sur la résolution des systèmes différentiels du second ordre avec conditions initiales et finale (cf.2.2) Leondes & Paines (1968) proposent de modifier la suite de fonctions \vec{z} (nous reprenons ici la notation du paragraphe 2.2) en introduisant un facteur correctif qui permet d'élargir le domaine d'initialisation de la méthode. La suite sera calculée par:

$$\vec{z}^{n+1}(t) = \vec{z}^n(t) + \alpha \cdot d\vec{z}(t)$$

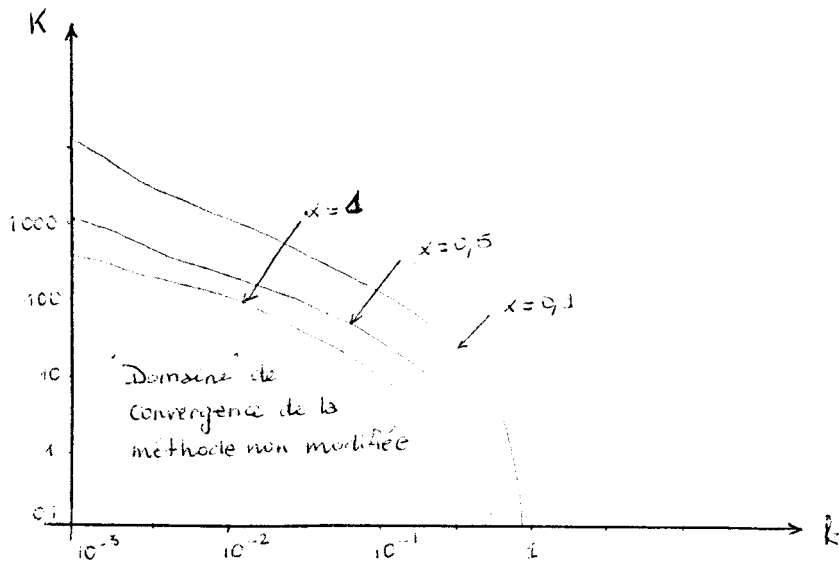
où α est un scalaire compris entre 0 et 1, constant égal à 1 à partir d'un certain N fixé. Il montrent que la convergence est encore assurée et ceci pour des fonctions initiales prises dans un plus grand voisinage de la solution, mais cette convergence n'est plus quadratique (tant que α est différent de 1). Ainsi l'augmentation du domaine de convergence se fera au détriment de la rapidité. En particulier on peut montrer que dans ce cas la suite modifiée vérifie, si \vec{G} est lipschitzienne:

$$\max_{i,t} | z_i^{n+1}(t) - z_i^*(t) | \leq \frac{k}{1-k} \cdot (1 + \alpha(2K+1) - k(1-\alpha)) \max_{i,t} | z_i^n(t) - z_i^{n-1}(t) |$$

où k et K sont les mêmes constantes que celles du théorème de Mc Gill (1963) cité en 1.2.

Les courbes $\frac{k}{1-k} \cdot (1 + \alpha(2K+1) - k(1-\alpha)) = 1$ sont données sur la figure suivante pour différentes valeurs de α .

Ainsi lorsque α tend vers 0, on peut constater que le domaine de convergence, ici "mesuré" par le couple (k, K) est plus grand mais la valeur de la contraction augmente, ce qui diminue la vitesse de convergence.



Miele (1974) suit le même raisonnement en réactualisant les fonctions $\vec{x}(t)$, $\vec{u}(t)$ et $\vec{\lambda}$ de la même façon, les variations $\delta(\cdot)$ étant obtenues par la résolution des équations (2.29) à (2.33). Une seconde approche consistera à trouver la meilleure initialisation possible à partir de données nominales. Cette dernière approche ne permet pas d'augmenter le domaine de convergence, nous la développerons ultérieurement.

Nous proposons dans ce paragraphe un nouveau point de vue: le fait de partir d'une trajectoire "proche de l'optimale" revient à dire que les erreurs tant sur la dynamique que sur les conditions finales sont faibles. Le rattrapage de ces erreurs se fera par une variation du premier ordre. Dans le cas où ces erreurs restent trop importantes, l'idée consiste à ne corriger qu'une partie de ces erreurs. Nous introduisons les matrices diagonales:

$$E_d = \text{diag}(\epsilon_{di}) \text{ où } \epsilon_{di} \in [0,1]$$

$$E_c = \text{diag}(\epsilon_{ci}) \text{ où } \epsilon_{ci} \in [0,1]$$

qui permettront de régler ces corrections.

Les variations modifiées seront alors

- pour la dynamique :

$$d \begin{pmatrix} \vec{x} - \vec{f}^* \\ \vec{\lambda} - \vec{h}^* \end{pmatrix} + E_d \cdot \begin{pmatrix} \vec{x} - \vec{f}^* \\ \vec{\lambda} - \vec{h}^* \end{pmatrix} = 0 \quad (2.89)$$

- pour les contraintes :

$$d \begin{pmatrix} \vec{x}(t_0) - \vec{x}_0 \\ \vec{\psi} \\ \vec{\lambda}(t_f) - \frac{\partial \Phi}{\partial \vec{x}} \end{pmatrix} + E_c \cdot \begin{pmatrix} \vec{x}(t_0) - \vec{x}_0 \\ \vec{\psi} \\ \vec{\lambda}(t_f) - \frac{\partial \Phi}{\partial \vec{x}} \end{pmatrix} = 0 \quad (2.90)$$

Le cas où $E_d = \mathbf{I}$ et $E_c = \mathbf{I}$ correspond au cas où les hypothèses de petites variations sont vérifiées.

Pour ne pas perdre l'avantage que donne la méthode par sa vitesse de convergence, les matrices devront être calculées de telle façon qu'à partir d'un certain nombre d'itérations on ait $E_d = \mathbf{I}$ et $E_c = \mathbf{I}$, où \mathbf{I} représente la matrice identité de dimension correspondante. En effet, les propriétés de convergence quadratique ne sont valables que pour ce cas.

Ces matrices permettront au début de l'algorithme de se rapprocher du domaine de convergence. Si nous considérons, de la même façon que Miele, le critère suivant:

$$Q = \int_{t_0}^{t_f} \left\| \vec{x} - \vec{f}^*(\vec{x}, \vec{\lambda}, t) \right\|^2 + \left\| \vec{\lambda} - \vec{h}^*(\vec{x}, \vec{\lambda}, t) \right\|^2 dt \\ + \left\| \vec{\psi}(\vec{x}_f, t_f) \right\|^2 + \left\| \vec{\lambda}^T(t_f) - \frac{\partial \Phi}{\partial \vec{x}}(\vec{x}_f, t_f, \vec{v}) \right\|^2$$

ce critère représente la mesure cumulée des erreurs par rapport aux conditions optimales. Une variation au premier ordre de $\delta \vec{x}$, $\delta \vec{\lambda}$ et $\delta \vec{v}$ donnera une variation δQ telle que

$$\delta Q = 2 \int_{t_0}^{t_f} \left(\frac{\partial}{\partial \vec{x}} (\vec{x} - \vec{f}^*) \right)^T \cdot d \left(\frac{\partial}{\partial \vec{x}} (\vec{x} - \vec{f}^*) \right) + \left(\frac{\partial}{\partial \vec{\lambda}} (\vec{\lambda} - \vec{h}^*) \right)^T \cdot d \left(\frac{\partial}{\partial \vec{\lambda}} (\vec{\lambda} - \vec{h}^*) \right) dt \\ + 2 \vec{\psi}^T \cdot d \vec{\psi} \\ + 2 \left(\vec{\lambda}^T - \frac{\partial \Phi}{\partial \vec{x}} \right) \cdot d \left(\vec{\lambda}^T - \frac{\partial \Phi}{\partial \vec{x}} \right)$$

Soit, en remplaçant les variations $d[\cdot]$ par $-E_d[\cdot]$ ou $-E_c[\cdot]$:

$$\begin{aligned} \delta Q = & - \int_{t_0}^{t_f} \left(\|\vec{x} - \vec{f}^*(\vec{x}, \vec{\lambda}, t)\|_{E_{d1}}^2 + \|\vec{\lambda} - \vec{h}^*(\vec{x}, \vec{\lambda}, t)\|_{E_{d2}}^2 \right) dt \\ & - \|\vec{\psi}(\vec{x}_{t_f}, t_f)\|_{E_{c1}}^2 - \|\vec{\lambda}^T(t_f) - \frac{\partial \Phi}{\partial \vec{x}}(\vec{x}_{t_f}, t_f, \vec{v})\|_{E_{c2}}^2 \end{aligned}$$

E_{d1} , E_{d2} , E_{c1} et E_{c2} représentent les matrices partielles correspondant respectivement aux variations de la dynamique de l'état, à celles de la dynamique de l'adjoint, à celles des contraintes sur l'état et à celles des contraintes finales sur l'adjoint. En restant dans les hypothèses de l'approximation du premier ordre, l'introduction de ces matrices permet donc d'assurer une diminution de l'erreur dans une "direction" donnée et d'atteindre le domaine de convergence de la méthode de quasilinearisation.

Il s'agit maintenant de calculer les suites E_d^N et E_c^N telles que:

$$\begin{aligned} \lim E_d^N &= \mathbf{I} \\ \lim E_c^N &= \mathbf{I} \end{aligned}$$

où E^N est la suite des matrices E au cours des itérations.

2.3.2. Calcul des matrices de contraction

Nous avons vu que le choix des matrices E conditionne la vitesse de convergence de l'algorithme. Ce choix est donc très important.

Afin d'estimer l'erreur par rapport aux conditions optimales, nous considérons les vecteurs $\vec{\rho}_d$ et $\vec{\rho}_c$ définis par

$$\begin{aligned} \rho_{di} &= \max_t \left| \dot{x}_i - f_i^*(\vec{x}(t), \vec{\lambda}(t), t) \right| & i = 1, nx \\ \rho_{di} &= \max_t \left| \dot{\lambda}_i - h_i^*(\vec{x}(t), \vec{\lambda}(t), t) \right| & i = nx+1, 2nx \\ \rho_{ci} &= \left| \psi_i(\vec{x}(t_f), t_f) \right| & i = 1, ncf \\ \rho_{ci} &= \left| \lambda_i(t_f) - \frac{\partial \Phi}{\partial x_i}(\vec{x}(t_f), t_f, \vec{v}) \right| & i = ncf+1, ncf+nx \end{aligned}$$

Soit les erreurs globales maximales autorisées données par R_{dmax} et R_{cmax} , on impose

$$\sum_{j=1}^{n_x} \max_t |x_j^{\circ} - f_j^*| + \sum_{j=1}^{n_x} \max_t |\lambda_j^{\circ} - h_j^*| < R_{dmax}$$

$$\sum_{j=1}^{n_{cf}} |\psi_j| + \sum_{j=1}^{n_x} \left| \lambda_j(t_f) - \frac{\partial \Phi}{\partial x_j} \right| < R_{cmax}$$

Nous poserons alors :

$$E_d = \text{diag} (\varepsilon_i) \text{ avec}$$

$$\varepsilon_i = \alpha_i \frac{R_d}{\rho_{di}} \quad \text{si } \frac{\rho_{di}}{R_d} \geq \alpha_i$$

$$\varepsilon_i = 1 \quad \text{si } \frac{\rho_{di}}{R_d} < \alpha_i$$

$$E_c = \text{diag} (\varepsilon_i) \text{ avec}$$

$$\varepsilon_i = \alpha_i \frac{R_c}{\rho_{ci}} \quad \text{si } \frac{\rho_{ci}}{R_c} \geq \alpha_i$$

$$\varepsilon_i = 1 \quad \text{si } \frac{\rho_{ci}}{R_c} < \alpha_i$$

Les coefficients α_i sont égaux à 1, ils pourront être modifiés lors des itérations pour affiner le réglage des variations. Cette méthode permet en quelque sorte de normaliser le système linéarisé en tenant compte des erreurs sur les dynamiques et les conditions finales. De plus, si les fonctions se rapprochent de la solution au cours des itérations successives la diminution des erreurs implique qu'à partir d'un certain nombre d'itérations $E_c = I$ et $E_d = I$.

Nous supposons alors atteint le domaine de convergence de la méthode de quasilinearisation classique. La convergence sera considérée comme acquise lorsque nous aurons

$$\frac{\rho_{di}}{R_d} < \tau \text{ et } \frac{\rho_{ci}}{R_c} < \tau \quad \forall i \text{ avec } \tau < 1$$

2.4. Recherche d'une "bonne" initialisation

Comme nous l'avons signalé précédemment, la méthode étant relativement sensible à l'initialisation, le choix des nominales est très important. Le problème de l'initialisation sera essentiellement l'estimation du vecteur \vec{v} et de la fonction adjointe nominale. Nous proposons ici, différentes méthodes.

La première méthode envisagée pour l'adjoint sera simplement l'intégration de l'équation différentielle à rebours à partir des conditions finales, \vec{x} , \vec{u} et \vec{v} étant donnés.

Une seconde méthode sera d'estimer l'adjoint initial et d'intégrer, en sens direct, le système différentiel en \vec{x} et $\vec{\lambda}$, la commande étant calculée par minimisation de l'hamiltonien. Le problème de l'initialisation de l'adjoint optimal pourra être résolu en utilisant la transformation commande-adjoint proposée par Dixon et al. (1981). Comme dans l'initialisation précédente, il sera nécessaire d'estimer le multiplicateur \vec{v} .

Une troisième méthode d'initialisation sera proposée (Yeo 1975). Elle repose sur la minimisation des erreurs sur l'adjoint et des conditions finales, étant donnés une commande et une trajectoire nominale. Cette dernière méthode donne un outil qui permettra une estimation du multiplicateur \vec{v} .

2.4.1. Initialisation par intégration

Soit une commande nominale $\vec{u}(t)$ et un multiplicateur \vec{v} . La trajectoire sera obtenue par intégration des équations d'état à partir des conditions initiales. Nous connaissons ainsi l'état à l'instant final. Supposons que nous ayons une bonne estimation des multiplicateurs de Lagrange, nous pouvons alors calculer les valeurs des adjoints à l'instant final. Les vecteurs adjoints seront alors le résultat de l'intégration en sens inverse du système différentiel adjoint.

Le défaut de ce type d'initialisation sera l'estimation du multiplicateur. Comme nous l'avons remarqué, il est nécessaire, pour se trouver dans le domaine de convergence de la méthode, d'avoir des valeurs assez proches d'une solution optimale. Cette condition sera souvent difficile à réaliser sur le multiplicateur. Par conséquent,

les adjoints étant intégrés en sens inverse à partir de ces multiplicateurs, l'initialisation sera d'autant plus mauvaise.

Nous envisageons une autre possibilité: supposons que nous remplaçons la commande par son expression en fonction de l'état et de l'adjoint obtenue par minimisation de l'hamiltonien. Si nous disposons d'une valeur initiale de l'adjoint, nous pourrions intégrer, en sens direct, le système différentiel formé par les équations d'états et les équations adjointes. Nous obtenons ainsi les nominales.

Cette méthode nécessite l'estimation de l'adjoint à l'instant initial. Dixon et al (1981) proposent une méthode simple pour cette estimation en établissant une transformation entre la commande et l'adjoint à l'instant initial.

Considérons une commande nominale. Pour vérifier l'optimalité, cette commande doit vérifier l'équation $\frac{\partial H}{\partial \vec{u}} = 0 \quad \forall t$. Nous aurons donc à l'instant initial

$$\begin{aligned} \frac{\partial H}{\partial \vec{u}} \Big|_{t_0} &= 0 \\ \frac{d^r}{dt^r} \left(\frac{\partial H}{\partial \vec{u}} \right) \Big|_{t_0} &= 0 \quad r = 2, 3, \dots \end{aligned}$$

Ces relations donnent alors des équations en $\vec{\lambda}(t_0)$ paramétrées par les dérivées successives de la commande \vec{u} à l'instant initial. Le choix de ces dérivées permet de calculer une valeur initiale de l'adjoint. L'estimation de ce dernier est donc remplacée par celle, a priori plus accessible, de la commande et de ses dérivées.

2.4.2. Initialisation de l'adjoint

Comme nous l'avons remarqué, le problème de l'initialisation vient surtout du multiplicateur de Lagrange et des adjoints. En effet, la commande et l'état pourront toujours être initialisés par des fonctions "physiquement" plausibles. Par contre nous n'avons aucun moyen de trouver une première approximation de $\vec{\lambda}$ et de \vec{v} .

Yeo (1975) propose une démarche qui permet de trouver, à partir d'une commande et d'un état donnés, un adjoint et un multiplicateur les "meilleurs" possibles.

Soient une commande donnée et l'état correspondant. Considérons la fonctionnelle:

$$E = \int_{t_0}^{t_f} \left(\left\| \vec{\lambda} + \frac{\partial H}{\partial \vec{x}} \right\|^2 + \left\| \frac{\partial H}{\partial \vec{u}} \right\|^2 \right) dt + \left\| \vec{\lambda}_f + \frac{\partial \Phi}{\partial \vec{x}_f} \right\|^2$$

et remarquons que E ne dépend que de $\vec{\lambda}$ et de \vec{v} (\vec{u} et \vec{x} étant fixés à priori). Pour obtenir un adjoint et un multiplicateur initiaux les meilleurs possibles, il est légitime de minimiser E par rapport à $\vec{\lambda}$ et \vec{v} . Nous avons alors à résoudre un problème de minimisation quadratique. Cette minimisation donne les équations :

$$\begin{aligned} \frac{d}{dt} \left(\vec{\lambda} + \frac{\partial H^T}{\partial \vec{x}} \right) - \frac{\partial \vec{f}}{\partial \vec{x}} \cdot \left(\vec{\lambda} + \frac{\partial H^T}{\partial \vec{x}} \right) - \frac{\partial \vec{f}}{\partial \vec{u}} \cdot \frac{\partial H^T}{\partial \vec{u}} &= 0 \\ \left(\vec{\lambda} + \frac{\partial H^T}{\partial \vec{x}} \right)_{t_f} + \left(\vec{\lambda} - \frac{\partial \Phi^T}{\partial \vec{x}} \right)_{t_f} &= 0 \\ \left(\vec{\lambda} + \frac{\partial H^T}{\partial \vec{x}} \right)_{t_0} &= 0 \\ \frac{\partial \vec{\psi}}{\partial \vec{x}} \cdot \left(\vec{\lambda} - \frac{\partial \Phi^T}{\partial \vec{x}} \right)_{t_f} &= 0 \end{aligned}$$

en posant

$$\begin{aligned} \vec{\varepsilon}(t) &= \left(\vec{\lambda} + \frac{\partial H^T}{\partial \vec{x}} \right) \\ \vec{\sigma} &= \left(\vec{\lambda} - \frac{\partial \Phi^T}{\partial \vec{x}} \right) \end{aligned}$$

et en remarquant que H dépend linéairement de l'adjoint, nous obtenons le système différentiel linéaire avec conditions aux extrémités suivant :

$$\begin{aligned} \dot{\vec{\varepsilon}} &= \frac{\partial \vec{f}}{\partial \vec{x}} \cdot \vec{\varepsilon} + \frac{\partial \vec{f}}{\partial \vec{u}} \cdot \frac{\partial \vec{f}^T}{\partial \vec{u}} \cdot \vec{\lambda} + \frac{\partial \vec{f}}{\partial \vec{u}} \cdot \frac{\partial \mathbf{L}^T}{\partial \vec{u}} \\ \dot{\vec{\lambda}} &= \vec{\varepsilon} - \frac{\partial \vec{f}^T}{\partial \vec{x}} \cdot \vec{\lambda} - \frac{\partial \mathbf{L}^T}{\partial \vec{x}} \\ \vec{\varepsilon}_{t_0} &= 0 \quad \vec{\varepsilon}_f + \vec{\sigma} = 0 \end{aligned}$$

$$\frac{\partial \vec{\psi}}{\partial \vec{x}_f} \cdot \vec{\sigma} = 0$$

$$\vec{\lambda}_f^T \cdot \left(\frac{\partial \phi}{\partial \vec{x}} + \vec{v}^T \cdot \frac{\partial \vec{\psi}}{\partial \vec{x}} \right) = \vec{\sigma}^T$$

Ce système sera résolu par une méthode de superposition :

pour $i=1, \dots, nx$, on calculera $\vec{\varepsilon}_f^i, \vec{\sigma}^i, \vec{v}^i$ solution de

$$\vec{\varepsilon}_f^i + \vec{\sigma}^i = 0$$

$$\frac{\partial \vec{\psi}}{\partial \vec{x}} \cdot \vec{\sigma}^i = 0$$

$$\vec{\sigma}^{iT} + \vec{v}^{iT} \cdot \frac{\partial \vec{\psi}}{\partial \vec{x}} = \vec{\lambda}_f^{iT} - \frac{\partial \phi}{\partial \vec{x}} \quad \text{où } \vec{\lambda}_f^{iT} = (0, \dots, 0, 1, 0, \dots, 0)$$

On intégrera le système à rebours à partir de $\vec{\lambda}_f^i$ et de $\vec{\varepsilon}_f^i$ ce qui donnera

$$\vec{\lambda}^i(t) \text{ et } \vec{\varepsilon}_0^i$$

La solution sera donnée par

$$\vec{\lambda}(t) = \sum_{i=0}^{nx} \alpha_i \cdot \vec{\lambda}^i(t)$$

$$\vec{v} = \sum_{i=0}^{nx} \alpha_i \cdot \vec{v}^i$$

avec

$$\begin{pmatrix} \vec{\varepsilon}_0^0 \cdot \vec{\sigma}^0 & \dots & \vec{\varepsilon}_0^{nx} \cdot \vec{\sigma}^{nx} \\ 1 & \dots & 1 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \dots \\ \alpha_{nx} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \dots \\ 1 \end{pmatrix}$$

3.MISE EN OEUVRE

3.1.Principe de l'application temps réel

Comme nous l'avons signalé, il sera envisagé d'appliquer cette méthode en temps réel. Avant de développer le programme qui permettra de tester la méthode, nous allons préciser ce que nous entendons ici par "application temps réel".

Cela consistera d'une manière générale à calculer en temps réel à intervalle de temps régulier une nouvelle commande optimale qui tiendra compte de l'état réel mesuré à l'instant courant.

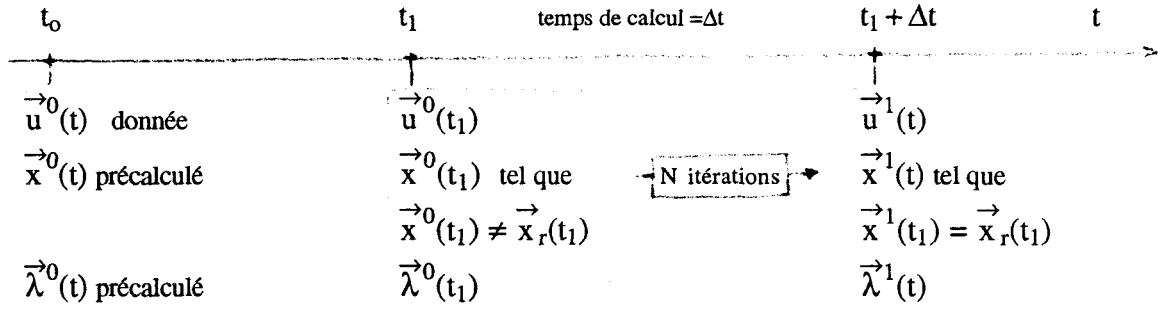
Avant de démarrer la trajectoire, une commande nominale optimale $\vec{u}^0(t)$ est calculée off-line à partir d'un modèle le plus précis possible. Cette commande sera appliquée au début de la trajectoire. A cette commande est associée un état $\vec{x}^0(t)$ et un adjoint $\vec{\lambda}^0(t)$ nominaux optimaux.

A un instant t_1 donné, on mesure l'état réel $\vec{x}_r(t_1)$ et on constate un décalage entre cet état et celui optimal théorique $\vec{x}^0(t_1)$ que l'on aurait dû atteindre. Nous supposons que ce décalage est dû soit à une erreur de modélisation, soit à des perturbations extérieures. La commande $\vec{u}^0(t)$ n'est donc pas exactement la commande optimale. Nous lançons alors à cet instant le calcul d'une nouvelle commande optimale $\vec{u}^*(t)$, avec $t \in [t_1, t_f]$, qui correspondra à $\vec{x}^*(t_1) = \vec{x}_r(t_1)$. Ce calcul se fera par la méthode de quasilinearisation avec les paramètres suivants:

- les fonctions qui initialisent l'algorithme seront $\vec{x}^0(t)$, $\vec{\lambda}^0(t)$, $t \in [t_1, t_f]$
- le modèle utilisé pour le calcul est le même que celui utilisé off-line
- le nombre N maximal d'itérations pour calculer la nouvelle commande est fixé.

N sera choisi petit. En supposant que l'algorithme converge rapidement, on sera assuré d'obtenir une commande qui minimisera le critère pour $t \in [t_1, t_f]$, avec la condition initiale $\vec{x}(t_1) = \vec{x}_r(t_1)$ et ceci au bout d'un temps de calcul relativement faible. Cette nouvelle commande sera alors appliquée au système.

Le schéma suivant résume le principe de guidage.



Prenons un exemple très simple. Dans cet exemple, la méthode de quasilinearisation n'est pas utile, mais cela permet d'illustrer la méthode d'application temps réel. Soit le système:

$$\begin{aligned} \dot{\vec{x}} &= \vec{u} \\ \dot{\vec{y}} &= \vec{v} \\ \dot{u} &= a_x - \rho \cdot \sqrt{u^2 + v^2} \cdot u \\ \dot{v} &= a_y - \rho \cdot \sqrt{u^2 + v^2} \cdot v - g \end{aligned}$$

On désire aller d'un état initial à $t_0 = 0$ à un état final à l'instant final $t_f = 400$ s donnés par:

$x_0 = 0$	$x_f = 1\ 000\ 0000\ \text{m}$
$y_0 = 0$	$y_f = 177\ 500\ \text{m}$
$u_0 = 0$	$u_f = 5700\ \text{m/s}$
$v_0 = 0$	$v_f = 0\ \text{m/s}$

en minimisant l'énergie:

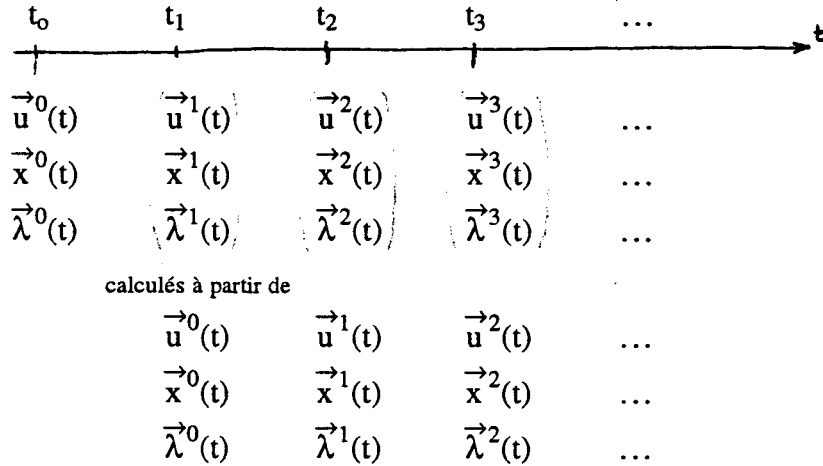
$$E = \frac{1}{2} \int_{t_0}^{t_f} (a_x^2 + a_y^2) \cdot dt$$

On suppose qu'il y a une erreur sur le modèle: la valeur de ρ pour le système réel sera 10^{-6} et 0 pour le modèle. Le problème d'optimisation modélisé est un problème linéaire quadratique. Il existe une solution analytique simple. Cette solution est calculée en une seule itération par notre méthode.

Nous avons supposé que Δt est négligeable et nous avons simulé le guidage pour 8 instants de réactualisation:

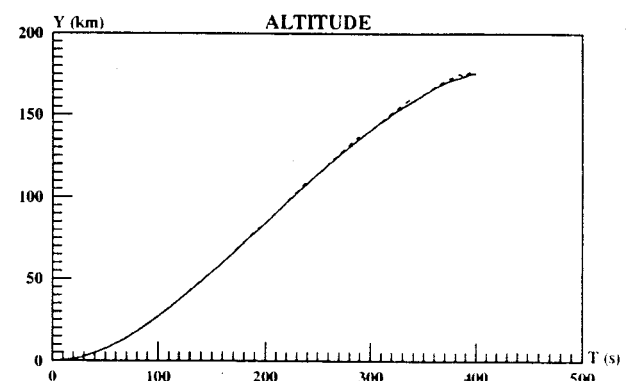
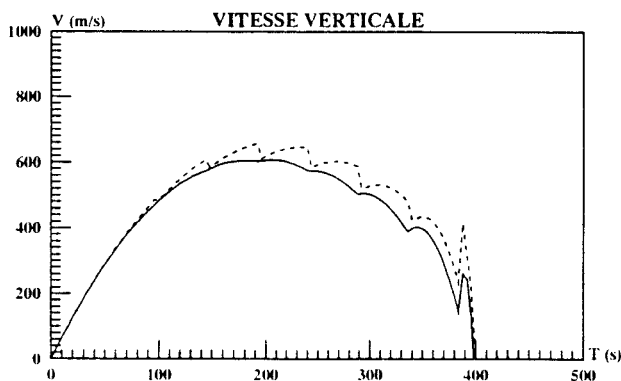
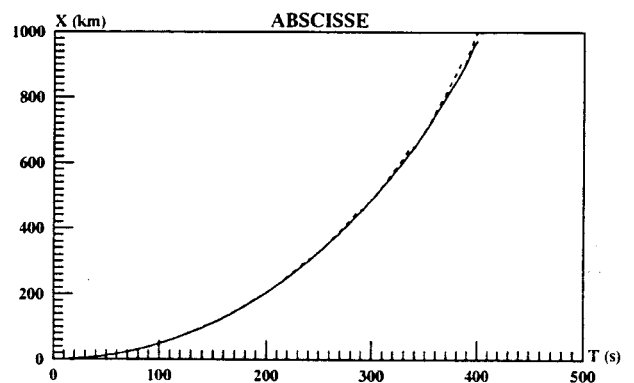
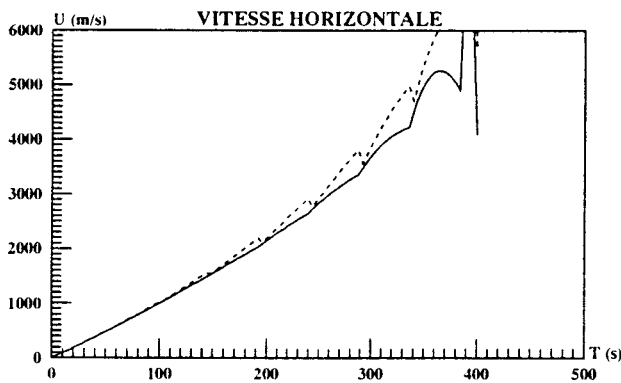
{48 s, 96 s, 144 s, 192 s, 240 s, 288 s, 336 s, 384 s}

Le schéma est le suivant:



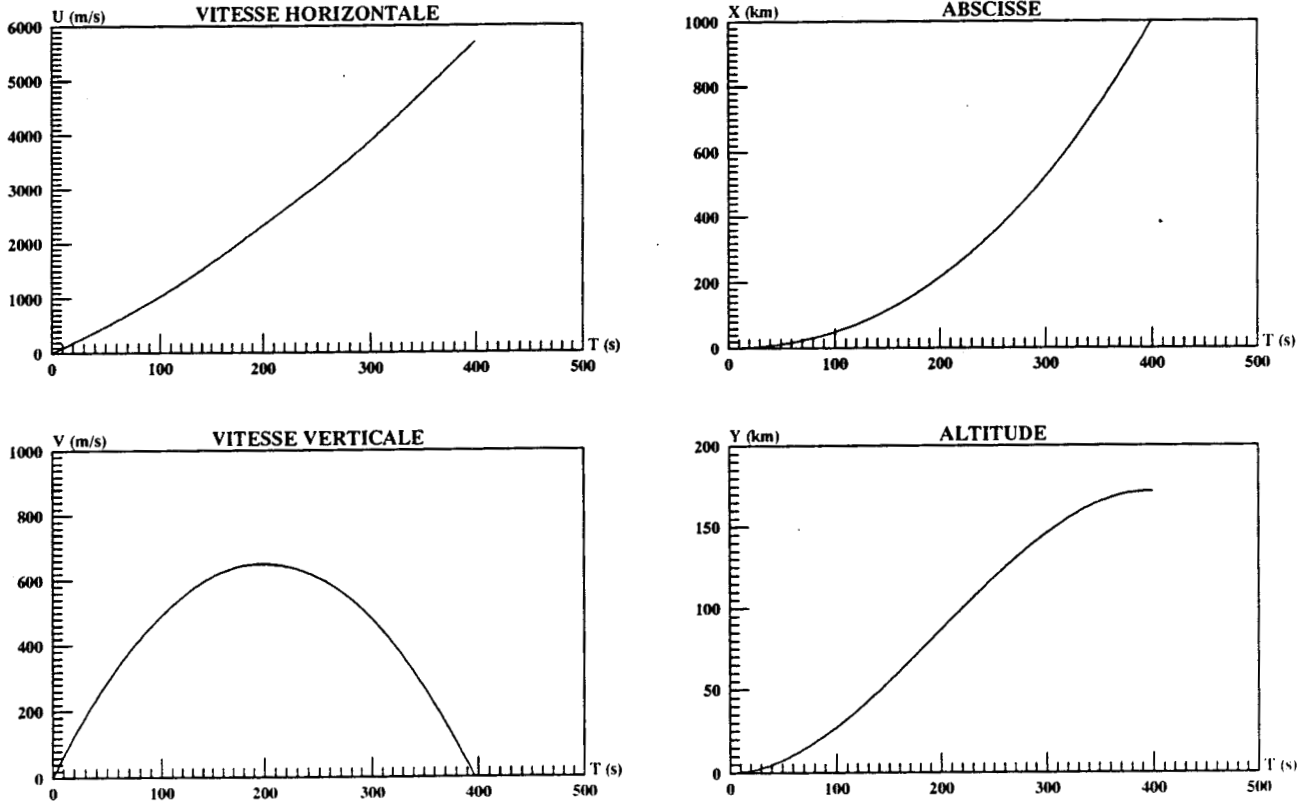
Les courbes obtenues sont données ci dessous. Les courbes en trait plein sont les courbes réelles, celles en trait pointillé sont les parties (entre deux instants de réactualisation) des courbes optimales recalculées à partir du modèle.

Remarquons que lorsque l'on se trouve très proche de l'instant final, les résultats ne sont pas satisfaisants en vitesse: les contraintes terminales ne sont pas vérifiées et la commande trop "chahuté" donne une vitesse peu réaliste. Il sera plus intéressant en fin de trajectoire d'appliquer une commande sans réactualisation qui permette d'atteindre les conditions finales de façon non optimale mais continue.

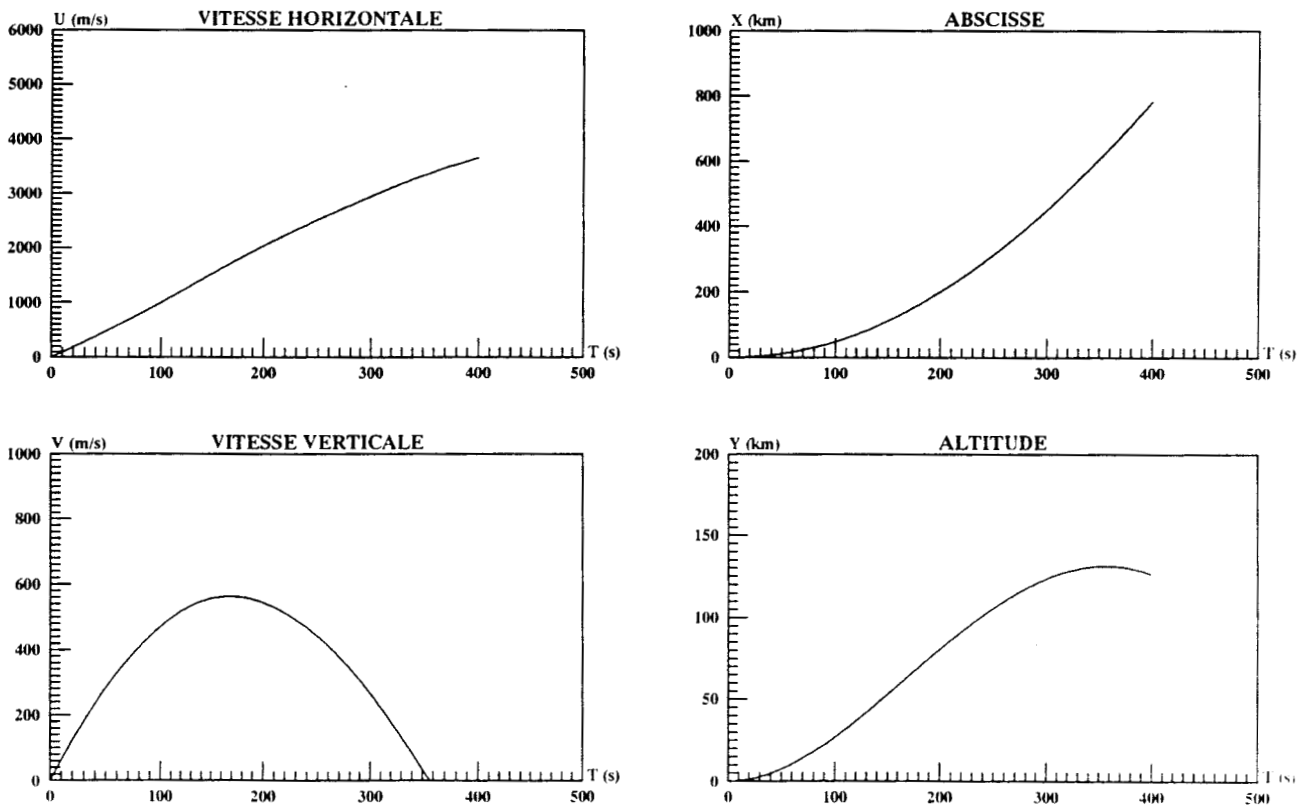


Pour comparaison, nous donnons sur la première figure les courbes optimales correspondant au modèle linéaire quadratique et sur la seconde figure les courbes réelles que l'on aurait obtenues si on avait appliqué la commande $\vec{u}^0(t)$ au système réel de t_0 à t_f , c'est à dire si l'on ne réactualise pas la commande.

- figure 1 -



- figure 2 -



3.2. Description du programme

3.2.1. Structure du programme

Nous avons mis en œuvre un programme général écrit en FORTRAN V afin de tester la méthode. La structure de ce programme est modulaire et offre la possibilité d'extensions à d'autres méthodes ou variantes de manière relativement simple. Au cours du développement du programme, nous avons en effet ajouté sans modification fondamentale de la structure la mise en œuvre de la méthode de tir direct.

Le programme permet donc le choix entre les méthodes suivantes :

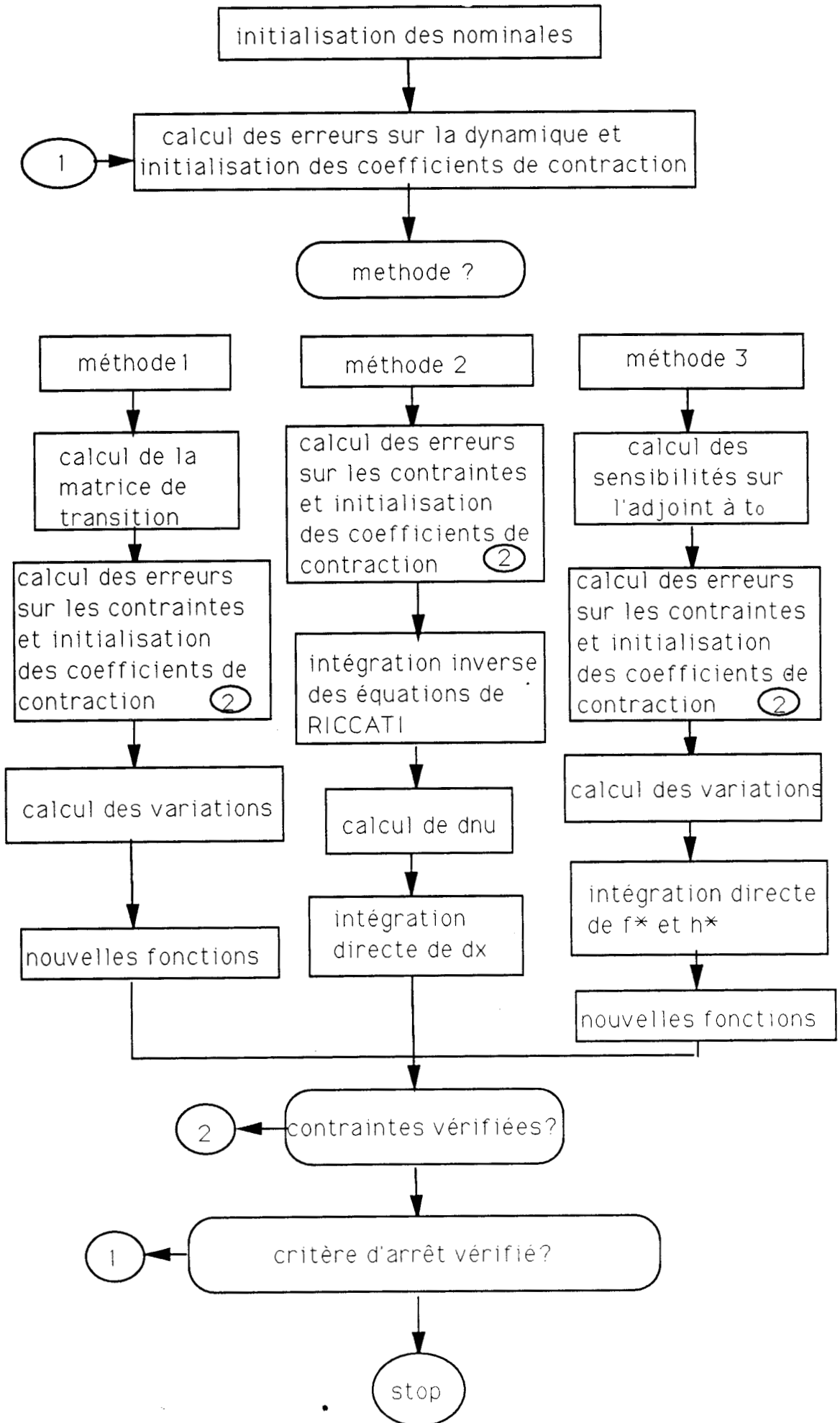
- Méthode de quasilinéarisation avec calcul de la matrice de transition
- Méthode de quasilinéarisation avec calcul des sensibilités directes
- Méthode de quasilinéarisation avec résolution des équations de RICCATI
- Méthode de tir par réactualisation de $\vec{\lambda}_0$ et \vec{v} .

Les fonctions nominales peuvent être initialisées suivant plusieurs modes :

- Initialisation par intégration directe de l'état et inverse de l'adjoint avec une commande nominale donnée*
- Initialisation par intégration directe de l'état et de l'adjoint avec minimisation de l'hamiltonien
- Initialisation par la méthode de Yeo
- Initialisation par fonctions données (lecture d'un fichier ou fonctions analytiques)

*La commande nominale sera soit donnée par une subroutine, soit lue à partir d'un fichier.

L'organigramme général est figuré page suivante.



3.2.2. Description des modules

Toutes les données utiles au problème et à la gestion de l'algorithme sont répertoriées dans un fichier dont la structure est fixée. Ces données sont lues au début de l'algorithme par la subroutine INIT.

Le système dynamique le critère et les contraintes sont définis par les subroutines suivantes:

- SUBF(T,X,U,F,Y)
définit le système dynamique et un vecteur de variables auxiliaires que l'utilisateur désire tracer.
- CMDEOPT(T,X,P,U)
définit la commande optimale en fonction de l'état X et de l'adjoint P
- CRITERE (T,X,U,L,G)
définit le critère, L étant l'intégrande et G la part finale du coût
- CONTRAINTE (T,X,U,W,V)
définit les contraintes d'égalité finale, W étant l'intégrande et V la part finale
- BORNE (T,X,Umin,Umax)
définit les bornes sur la commande fonctions de l'état.

De plus, il est possible de définir le système adjoint et le système linéarisé de façon analytique (s'ils ne sont pas donnés, le calcul est effectué numériquement). Ces fonctions sont définies dans les subroutines :

- SUBADJ(T,X,U,P,H)
définit le système adjoint
- SYSLINEA (T,X,P,R)
définit la matrice R du système linéarisé.

Pour l'initialisation des fonctions, on donne les subroutines

- CMDENOM (T,X,U)
(mode d'initialisation 3)
- NOMINALE (T,X,P,F,H)
(mode d'initialisation 4)

Les variables sont en double précision. Les résultats sont écrits dans un fichier de sortie par la subroutine RESU.

3.2.3. Choix des données et des méthodes

Les données sont précisées dans un fichier data structuré de la façon suivante:

DIMENSIONS

n_x = dimension du vecteur état
 n_u = dimension du vecteur commande
 n_{ce} = nombre de contraintes d'égalité
 n_{ci} = nombre de contraintes d'inégalité

COEFFICIENTS DE NORMALISATION

$\Delta x(n_x)$ = sur l'état
 $\Delta u(n_u)$ = sur la commande
 ΔJ = sur le critère
 $\Delta C(n_{ce}+n_{ci})$ = sur les contraintes

VARIATIONS POUR LE CALCUL DES DERIVEES PARTIELLES

E_x = coefficient de variation pour l'état (en % de Δx)
 E_u = coefficient de variation pour la commande (en % de Δu)
 E_{tps} = coefficient de variation pour le temps

COEFFICIENT DE PERTURBATION DE L'ADJOINT INITIAL

$E\lambda_1$ = donné pour le calcul des sensibilités directes (en % de $\lambda(t_1)$)

DONNEES TEMPORELLES

t_i = instant initial
 t_f = instant final
 dt = intervalle de discretisation pour l'intégration

NOMBRE DE BORNES SUR LA COMMANDE

$n_{bu}(n_u)$ =

MODE D'INITIALISATION

mode = (1 pour reprise de résultats du gradient projeté, 2 pour reprise, 3 pour la première initialisation, 4 pour la seconde, 5 pour la troisième, 6 pour des fonctions données par sous-routine nominale)

NOMBRE MAXIMAL D'ITERATION

n_{biter} =

COEFFICIENT D'ERREURS MAXIMALES POUR LE CALCUL DES POIDS

$\alpha(2.n_x+n_{ce}+n_{ci})$ =

MULTIPLICATEURS NOMINAUX

$v(n_{ce}+n_{ci})$ =

CONDITIONS INITIALES

$x_1(nx)$ =

$\lambda_1(nx)$ = utilisé pour la méthode d'initialisation en mode 4

CRITERE D'ARRET

τ =

CHOIX DES METHODES

algorithme = (1 pour la méthode de la matrice de transition,
2 pour le calcul des sensibilités directe,
3 pour l'intégration des équations de RICCATI,
4 pour la méthode de tir)

intégration = (1 pour intégration par Runge-Kutta-Gill,
2 pour intégration par Adams-Moulton)

expression = (0 pour le calcul numérique de l'adjoint,
1 pour le calcul analytique par la subroutine ADJOINT)

Remarques :

Comme nous l'avons signalé, la méthode par intégration des équations de Riccati est soumise à des conditions parfois difficile à assurer. L'algorithme risque dans ce cas d'être interrompu. La méthode des sensibilités directe donnera des résultats plus ou moins précis selon le pourcentage de perturbation de la valeur initiale de l'adjoint, il sera donc nécessaire d'en avoir une bonne estimation.

3.3.Exemples numériques

Pour illustrer les différentes possibilités de l'algorithme ainsi que pour tester l'influence des coefficients multiplicateurs sur l'évolution des fonctions au cours des itérations, nous avons traité plusieurs exemples simples. Les deux premiers exemples ont une solution analytique. Nous pourrions donc savoir exactement si la solution numérique obtenue est la solution optimale. Le dernier exemple, pris dans la littérature, nous permettra de comparer nos résultats avec ceux obtenus par une autre méthode de quasilinearisation.

Le premier exemple considéré est un système simple à faible non linéarité et commande bornée. Cet exemple nous permettra de vérifier le bon comportement de l'algorithme vis à vis du traitement des bornes et de donner des premiers repères pour la gestion des matrices de contraction.

Le second exemple est celui du décollage du L.E.M. (Bryson 1975). Cet exemple est non linéaire, mais la solution analytique est connue.

Enfin, nous étudierons un exemple plus complet emprunté à la littérature. Ce dernier exemple nous permettra de comparer nos résultats avec ceux obtenus par une méthode de quasilinearisation développée par Yeo (1976).

3.3.1. Exemple préliminaire

Considérons le système très simple du double intégrateur avec une non linéarité due à un frottement fluide:

$$\begin{aligned}\dot{x} &= v \\ \dot{v} &= u - \rho \cdot v^2\end{aligned}$$

Nous imposons des conditions initiales et finales :

$$\begin{array}{ll}t_0 = 0 & t_f = 10 \text{ s} \\ x_0 = 0 & x_f = 200 \\ v_f = 0 & v_f = -10\end{array}$$

et des bornes fixes sur la commande

$$- u_M \leq u(t) \leq u_M$$

Le critère à minimiser sera

$$J = \int_{t_0}^{t_f} u^2 dt$$

Pour $\rho = 0$, ce problème est un problème linéaire quadratique. Nous allons étudier ce cas très simple pour différentes valeurs de la borne u_M . Bien que ce cas soit relativement trivial, il permettra d'illustrer de façon claire le comportement de l'algorithme.

Solution optimale:

La solution optimale de ce problème est analytique, elle est donnée en annexe 2. Les adjoints sont des fonctions simples: λ_x est une fonction constante, λ_v est une fonction linéaire du temps. Lorsque les contraintes ne sont pas actives, la vitesse et l'abscisse sont données par:

$$x(t) = \frac{v_x}{6} \cdot t^3 - (v_x t_f + v_v) \cdot \frac{t^2}{2} + v_o \cdot t + x_o$$

$$v(t) = \frac{v_x}{2} \cdot t^2 - (v_x t_f + v_v) \cdot t + v_o$$

et la commande optimale par:

$$u(t) = -\lambda_v(t)$$

Pour le cas où la commande n'est pas bornée, la solution est immédiate. Pour le cas où la commande est limitée par u_M , nous aurons une ou deux commutations selon la valeur de la borne (si les contraintes deviennent actives pendant l'intervalle de temps).

Initialisation:

L'initialisation choisie est la suivante (cette initialisation est arbitraire):

$$x(t) = 10 \cdot (1 - \cos(\pi t / 10)) \quad \lambda_x(t) = 1$$

$$v(t) = 10 \cdot \sin(\pi t / 10) \quad \lambda_v(t) = 1$$

Résultats numériques:

Nous avons effectué des essais pour $u_M \in \{10, 12, 14, 20\}$. Pour $u_M \in \{10, 12\}$ les deux bornes sont actives, pour $u_M = 14$ seule la borne inférieure est active et le cas $u_M = 20$ représente le cas non contraint.

Les résultats sont figurés sur les tableaux suivants. Nous donnons, dans le premier, les valeurs optimales des multiplicateurs et des instants de commutation (s'il y a commutation) et dans le second, les valeurs obtenues par l'algorithme ainsi que le nombre d'itération avant la convergence.

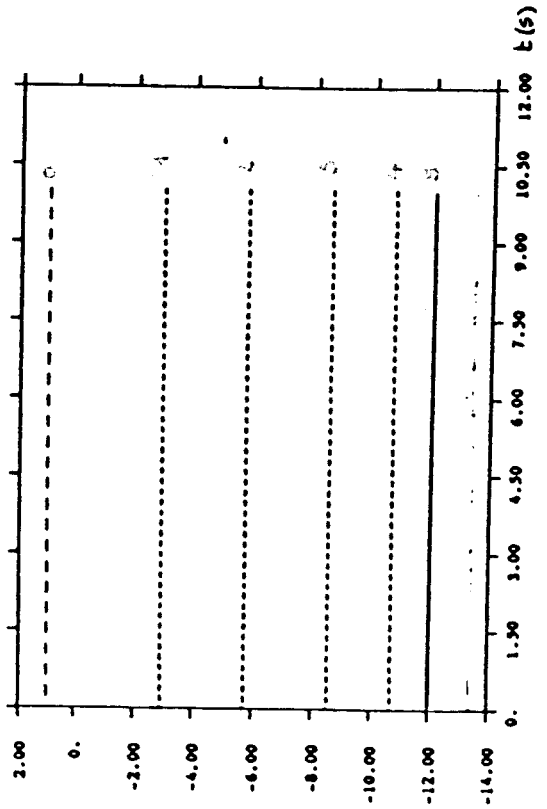
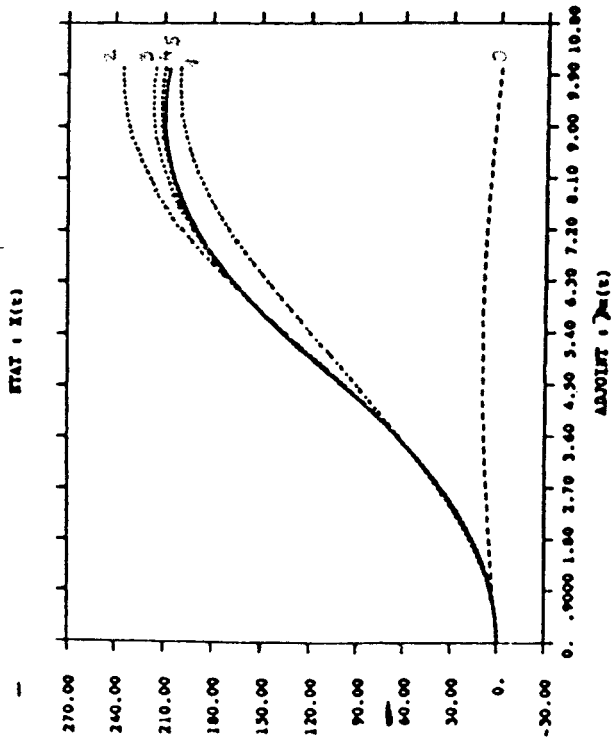
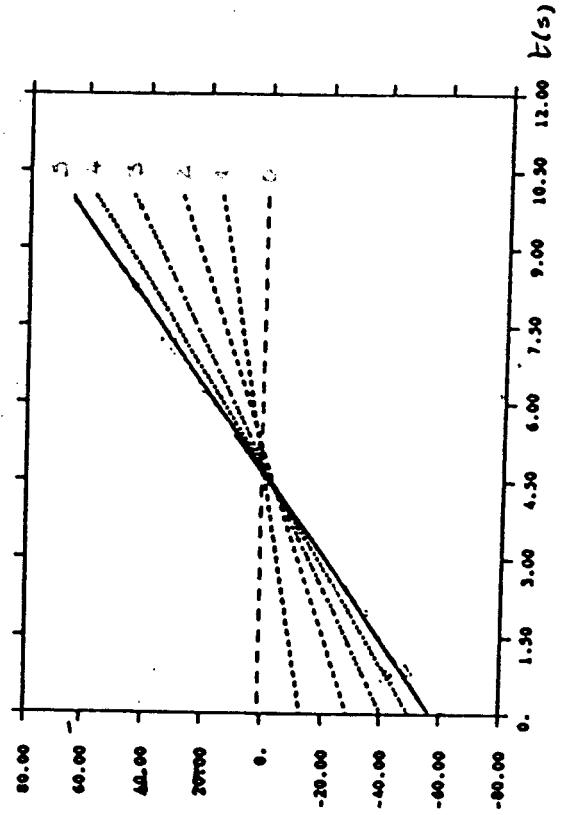
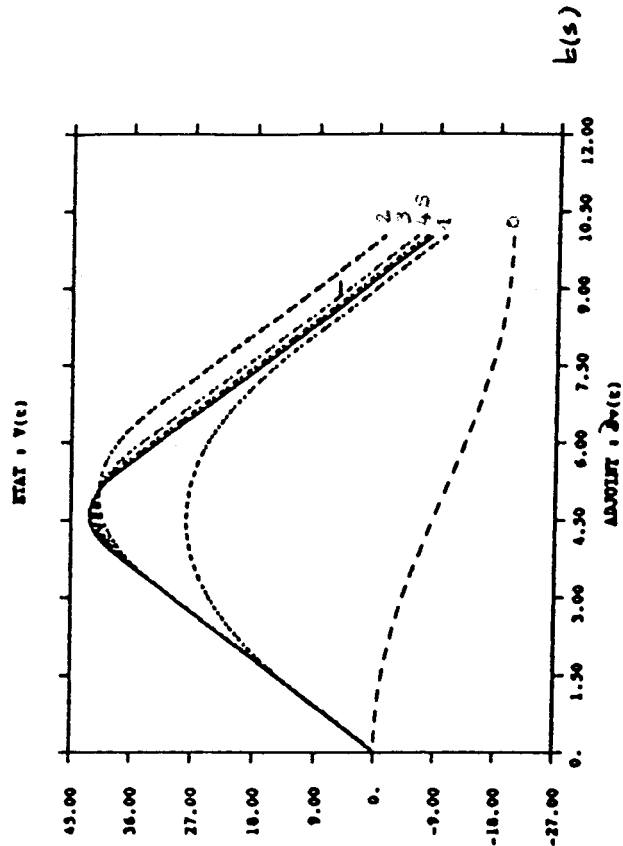
Il n'y a pas de matrices de contraction. Nous avons $E_d = I$ et $E_c = I$. Le critère d'arrêt sera: $\max_t | \dot{x} - v | + \max_t | \dot{v} - u | + \max_t | \dot{\lambda}_x | + \max_t | \dot{\lambda}_v + \lambda_x(t) | < 0,01$

et lorsque l'erreur sur les conditions finales sur l'état et l'adjoint sont inférieures à 1%.

u_M	valeurs optimales			
	v_1	v_2	t_1	t_2
10.00	-11.55	63.51	3.65	5.37
12.00	-3.460	18.72	1.12	8.04
14.00	-3.051	16.35	-	9.23
20.00	-3	16	-	-

u_M	valeurs obtenues				nombre d'itérations
	v_1	v_2	t_1	t_2	
10.00	-12.05	63.6	3.8	5.4	5
12.00	-3.420	18.62	1.1	8.	3
14.00	-2.996	16.31	-	9.3	2
20.00	-2.999	15.99	-	-	1

Les résultats obtenus montrent sur cet exemple la bonne convergence de l'algorithme et le bon comportement vis à vis du traitement des bornes sur la commande. Pour illustrer l'évolution des fonctions au cours des itérations, nous donnons, pour le premier cas, les courbes obtenues au cours des 5 itérations.



Nous considérons maintenant une non linéarité avec $\rho = 0,01$. Dans le but de tester la sensibilité de l'algorithme à l'initialisation, dans ce cas simple, plusieurs essais ont été effectués pour différentes initialisations.

Initialisations:

1) fonctions optimales obtenues avec $\rho = 0$

2) adjoints nuls, état par intégration directe avec

$$\begin{aligned} u(t) &= 1 \\ x(t) &= t^2 & \lambda_x(t) &= 0 \\ v(t) &= 2.t & \lambda_v(t) &= 0 \end{aligned}$$

3) intégration des adjoints à partir de $\lambda_x(t_f) = -1$ et $\lambda_v(t_f) = 10$ et fonctions sinusoidales pour l'état

$$\begin{aligned} u(t) &= -\lambda_v(t) \\ x(t) &= 10.(1-\cos(0,314.t)) & \lambda_x(t) &= -1 \\ v(t) &= 10.\sin(0,314.t) & \lambda_v(t) &= t \end{aligned}$$

4) intégration inverse des adjoints avec $\lambda_x(t_f) = \frac{1}{2}$ et $\lambda_v(t_f) = 1$ et intégration directe de l'état avec $x(t_0) = 0$ et $v(t_0) = 0$

$$\begin{aligned} u(t) &= -\lambda_v(t) \\ x(t) &= \frac{1}{12}.t^3 + 3.t^2 & \lambda_x(t) &= \frac{1}{2} \\ v(t) &= \frac{1}{4}.t^2 + 6.t & \lambda_v(t) &= -\frac{1}{2}.t + 6 \end{aligned}$$

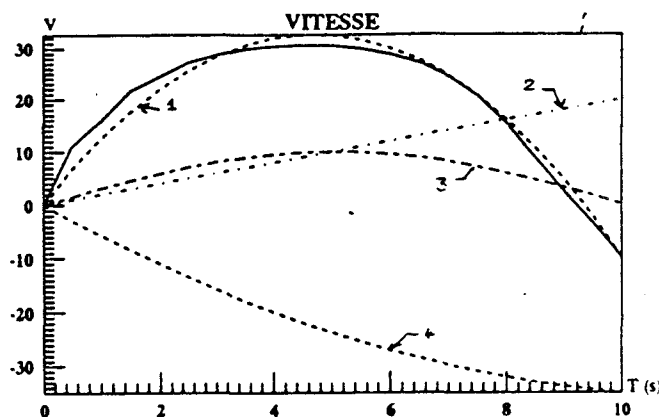
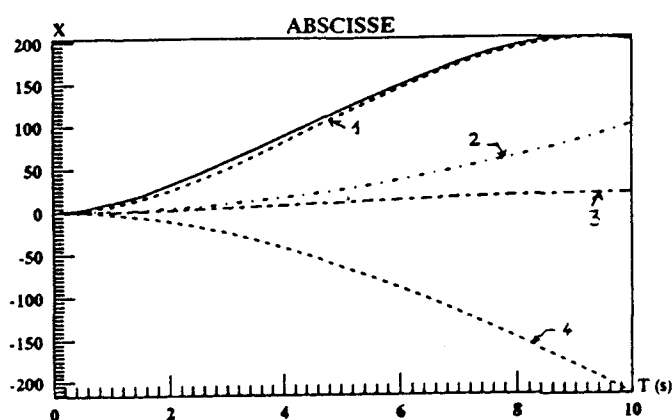
résultats numériques:

Tous résultats obtenus sont figurés sur le tableau suivant. La convergence sera considérée comme atteinte lorsque toutes les erreurs sont inférieures à 1%.

Dans ce cas, nous n'avons pas de solution analytique. Cependant, les erreurs sur la dynamique et sur les contraintes finales étant quasiment annulés, nous pouvons supposer que la solution est atteinte. Par ailleurs, nous pouvons remarquer que pour un même cas, on obtient les mêmes courbes quelque soit l'initialisation choisie, ce qui pourrait confirmer l'optimalité de la solution.

Nous avons figuré sur la figure page suivante les initialisations ainsi que la courbe obtenue à la convergence pour le cas correspondant à $u_M = 12$.

INITIALISATION	u_M	VALEURS OBTENUES		NOMBRE D'ITERATIONS	J
		v_1	v_2		
1	12	-7,15707	26,65721	3	496
	14	-6,54872	23,26344	3	528
	20	-5,93291	20,55912	3	546
2	12	-7,15707	26,65721	4	496
	14	-6,54872	23,26344	4	528
	20	-5,93291	20,55912	4	546
3	12	-7,15707	26,65721	3	496
4	12	-7,15707	26,65721	4	496



— résultat obtenu à la convergence

- - - initialisation n

Nous constatons que l'algorithme converge pratiquement aussi rapidement dans le cas sans contrainte ($u_M = 20$) que dans les autres cas., ce qui montre le bon comportement du programme vis à vis du traitement des bornes sur la commande.

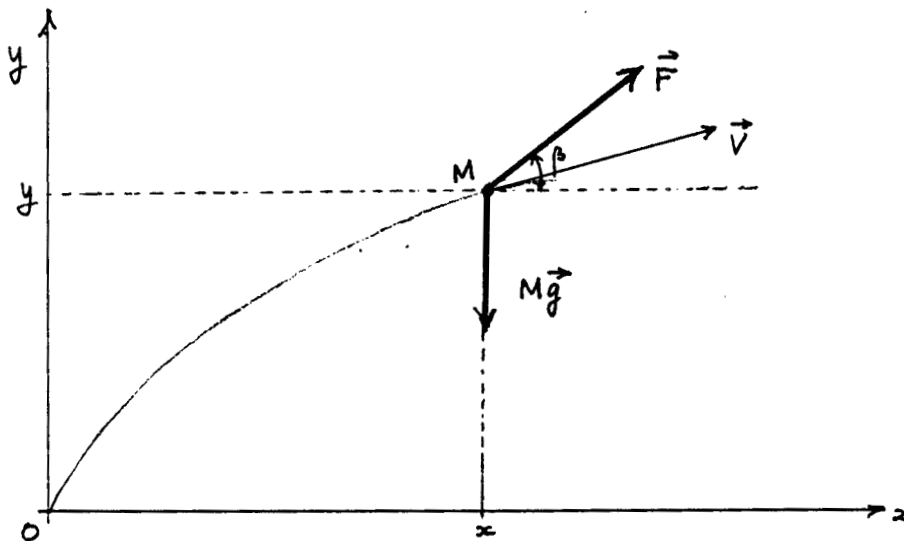
Nous pouvons remarquer de plus que pour un système faiblement non linéaire, l'algorithme converge rapidement même si l'initialisation est éloignée de la solution. Ce premier résultat indique qu'il est envisageable sur des cas simples d'utiliser cette méthode pour le guidage. Il s'agira de traiter des exemples plus complets pour le confirmer.

3.3.2. Décollage du L.E.M. lunaire

Considérons le problème suivant : on désire faire décoller le L.E.M. lunaire pour atteindre une position d'orbite à une altitude donnée avec une vitesse finale horizontale et maximale. La poussée du L.E.M. sera de module constant, de direction variable.

Equations d'état et définition du problème

Le L.E.M. est soumis à la force de pesanteur lunaire et à une force de propulsion de module constant et de direction variable.



β est l'angle entre la direction de poussée et l'horizontale, ce sera la commande.

Les équations d'état et les conditions terminales sont les suivantes:

$$\begin{aligned} \dot{u} &= \frac{F}{M} \cos\beta \\ \dot{v} &= \frac{F}{M} \sin\beta - g \\ \dot{x} &= u \\ \dot{y} &= v \\ u(t_f) &= u_f \text{ maximum} \\ v(t_f) &= 0 \\ x(t_f) &= \text{quelconque} \\ y(t_f) &= y_f \end{aligned}$$

Solution optimale

Les équations adjointes sont immédiates:

$$\begin{aligned}\lambda_u(t) &= -1 \\ \lambda_v(t) &= -v_y \cdot t + (v_y \cdot t_f + v_v) \\ \lambda_x(t) &= 0 \\ \lambda_y(t) &= v_y\end{aligned}$$

et la commande optimale sera donnée par une loi linéaire tangente:

$$\beta^* = \text{Arctg} \left[\frac{\lambda_v}{\lambda_u} \right]$$

soit
$$\beta^* = \text{Arctg} [v_y \cdot t - (v_y \cdot t_f + v_v)]$$

Le problème a une solution analytique (Bryson & Ho, 1975). En utilisant β comme variable d'intégration et en posant $\text{tg}\beta = \text{tg}\beta_0 - kt$, nous obtenons les fonctions optimales :

$$\begin{aligned}u(t) &= \frac{a}{k} f(\beta) \\ v(t) &= \frac{a}{k} \cdot (\sec\beta_0 - \sec\beta) - gt \\ x(t) &= \frac{a}{k^2} \cdot (\sec\beta_0 - \sec\beta - \text{tg}\beta \cdot f(\beta)) \\ y(t) &= \frac{a}{k^2} \cdot (\sec\beta_0 \cdot (\text{tg}\beta_0 - \text{tg}\beta) - \text{tg}\beta \cdot (\sec\beta_0 - \sec\beta) - f(\beta)) - \frac{1}{2} gt^2\end{aligned}$$

où
$$f(\beta) = \log \left(\frac{(\text{tg}\beta_0 + \sec\beta_0)}{(\text{tg}\beta + \sec\beta)} \right)$$

et
$$a = \frac{F}{M}$$

Les paramètres β_0 et k seront déterminés par les conditions terminales:

$$v(t_f) = 0 \text{ et } y(t_f) = y_f$$

Avec $y_f = 177$ km, nous aurons:

$$\begin{aligned}\beta_0 &= 0,75 \text{ rad} \\ k &= 0,00268\end{aligned}$$

ce qui donne pour l'adjoint: optimal correspondant:

$$\lambda_x(t) = 0$$

$$\lambda_y(t) = - 0,00268$$

$$\lambda_u(t) = - 1$$

$$\lambda_v(t) = 0,00268 t - 0,9315$$

Initialisation :

Nous avons choisi comme initialisations les cas suivants:

- 1) fonctions optimales pour $y_f = 250$ km, $(v_y, v_v) = (0,0)$
- 2) fonctions optimales pour $y_f = 326,8$ km, $(v_y, v_v) = (0,0)$
- 3) fonctions optimales pour $y_f = 406,6$ km, $(v_y, v_v) = (0,0)$
- 4) état approché par des fonctions linéaire et sinusoidales telles que les conditions finales soient presque vérifiées et et adjoint non optimal:

$$x(t) = 14 t$$

$$\lambda_x(t) = 0$$

$$y(t) = 650\sin(\pi t / 400)$$

$$\lambda_y(t) = 1$$

$$u(t) = 14$$

$$\lambda_u(t) = - 1$$

$$v(t) = 82760(1-\cos(\pi t / 400)) \quad \lambda_v(t) = - t$$

- 5) état nominal donné par des fonctions polynomiales et adjoint optimal

$$x(t) = \frac{1}{2} t^2$$

$$y(t) = \frac{1}{2} t^2$$

$$u(t) = t$$

$$v(t) = t$$

- 6) état nominal précédent et adjoint non optimal donné à l'exemple 4

Résultats numériques:

Les résultats numériques sont résumés sur le tableau page suivante.

Nous pouvons remarquer une plus grande sensibilité de l'algorithme par rapport aux adjoints. Cela s'explique par le fait que l'expression de la commande optimale dépend de ces adjoints, une erreur importante sur ceux-ci "fausse" donc de façon importante la commande initiale. Il sera donc important, sur des exemples plus réalistes, d'avoir une bonne estimation de ces adjoints.

Nous avons constaté une divergence dans le dernier cas traité. Afin de tester l'efficacité des modifications apportées par les matrices de contraction, nous avons effectué un nouvel essai avec:

$$E_c = \text{diag} (\varepsilon_i) \text{ avec}$$

$$\varepsilon_i = 0,1 \frac{R_c}{\rho_{ci}} \quad \text{si } \frac{\rho_{ci}}{R_c} \geq 0,1$$

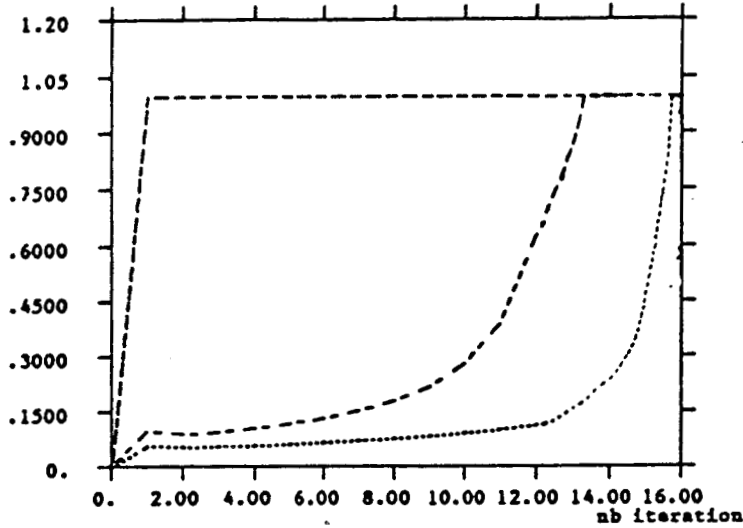
$$\varepsilon_i = 1 \quad \text{si } \frac{\rho_{ci}}{R_c} < 0,1$$

L'algorithme converge alors en 16 itérations, ce qui confirme dans ce cas particulier, l'augmentation du domaine de convergence.

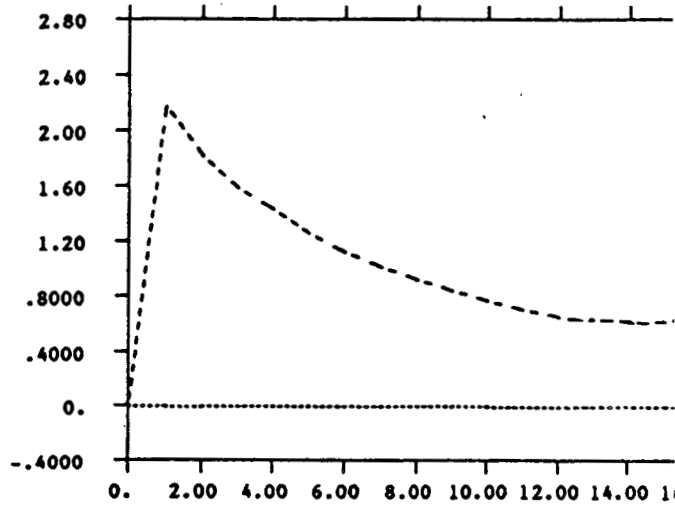
essai n°	données initiales	nombre d'itération pour la convergence
1	v = 0 optimum pour h = 250 km	3
2	v = 0 optimum pour h = 360,8 km	3
3	v = 0 optimum pour h = 406,6 km	4
4	v = 0	14
5	v = 0	2
6	v = 0	DV
7	même initialisation que pour 4 matrice de contraction sur les contraintes	16

L'évolution des courbes pour ce dernier cas est donné ci dessous. Nous avons figuré la commande et la trajectoire obtenues au cours des itérations ainsi que l'évolution des multiplicateurs et des coefficients de la matrice de contraction.

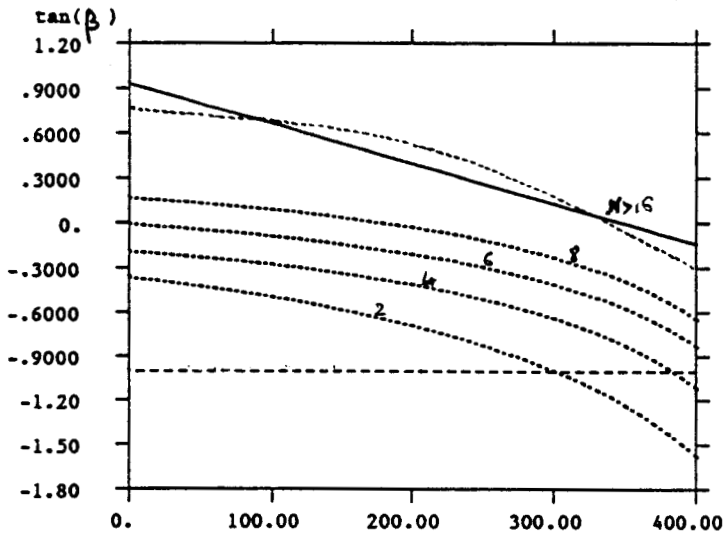
POIDS SUR CONTRAINTES



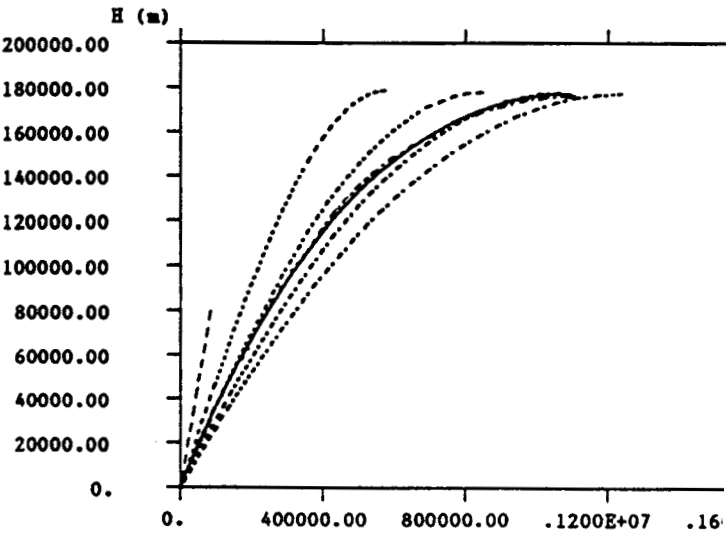
MULTIPLICATEURS



COMMANDE : TAN(β)



TRAJECTOIRE



----- initialisation
..... itérations
———— optimum

Ces différents essais mettent en évidence la grande sensibilité de convergence vis à vis de l'adjoint nominal. En effet, l'algorithme diverge pour un adjoint éloigné de l'optimum, ceci même lorsque l'état est donné par la trajectoire optimale. L'utilisation du coefficient multiplicateur permet alors d'élargir le domaine de convergence, le choix de α conditionne alors le nombre d'itérations nécessaires avant la convergence.

Au vu de ces essais, nous constatons la nécessité d'avoir un vecteur adjoint nominal le plus proche possible de l'optimum. Etant donné le manque de connaissance dont nous disposons pour estimer l'adjoint, cette condition sera parfois difficile à réaliser. Il sera donc intéressant dans le cas où nous ne disposons pas d'une commande proche de l'optimum d'utiliser le mode d'initialisation qui minimisera l'erreur sur l'adjoint.

3.3.3. Exemple de Yeo

Nous avons étudié un exemple plus complet pris dans la littérature (Yeo 1976). Il s'agit d'un système non linéaire à deux commandes soumises à des contraintes non linéaires dépendant de l'état.

Les équations du système sont les suivantes:

$$\begin{aligned}\dot{x}_1(t) &= x_2 \\ \dot{x}_2(t) &= (1-x_1^2)x_2 - x_1 + u_1 + u_2\end{aligned}$$

Le problème consistera à minimiser le critère :

$$J = \int_{t_0}^{t_f} (x_1^2 + x_2^2 + u_1^2 + 2 \cdot u_2^2) \cdot dt$$

en respectant les contraintes initiales et finales :

$$\begin{aligned}R &\leq 0 \quad \text{avec} \\ R &= x_2^2 + u_1^2 - 1.3 \\ x_1(t_0) &= 1 & x_1(t_f) &= 0 \\ x_2(t_0) &= 0 & x_2(t_f) &= 0 \\ t_f &= 1,73\end{aligned}$$

Les valeurs optimales du multiplicateur de lagrange et de l'adjoint optimal à l'instant initial sont données par Yeo:

$$\begin{aligned} v_1 &\approx +6 \\ v_2 &\approx -5,4 \\ \lambda_1(0) &\approx 9,5 \\ \lambda_2(0) &\approx 5,5 \end{aligned}$$

Initialisations

Nous nous proposons tester la convergence sur différentes initialisation et de comparer nos résultats avec ceux obtenus par Yeo.

Les modes d'initialisation sont les suivants:

1) intégration directe de l'état et inverse de l'adjoint à partir de v_1 et v_2 donnés, la commande étant donnée par celle définie par Yeo (approximation linéaire de la solution):

$$\begin{aligned} u_1(t) &= -0,25 & 0 \leq t \leq 0,2.t_f \\ & -0,25 + 1,25.(t-0,2.t_f)/0,6.t_f & 0,2.t_f \leq t \leq 0,8.t_f \\ & 1 & 0,8.t_f \leq t \leq t_f \\ u_2(t) &= u_1(t) \end{aligned}$$

2) intégration directe de l'état et de l'adjoint.

3) intégration directe de l'état et adjoint donné par

$$\begin{aligned} \lambda_1(t) &= k \\ \lambda_2(t) &= k t / t_f \end{aligned}$$

Résultats numériques:

mode d'initialisation	Nombre d'itérations	(résultat de Yeo)
1 $v = v^*$	2	
1 $v = 0$	DV	
2 $\lambda(0) = \lambda^*(0)$	2	
2 $\lambda(0) = 0$	10	
3 $k = -8$	DV	DV
3 $k = -4$	7	7
3 $k = 0$	7	7
3 $k = 0,4$	8	7
3 $k = 0,8$	14	DV
3 $k = 1$	19	DV

CONCLUSION

Après avoir analysé la quasilinéarisation, nous avons mis en évidence les caractéristiques de cette méthode et nous avons étudié les différentes variantes de l'algorithme. Nous avons proposé dans une démarche unifiée où la commande est donnée par une équation implicite, fonction de l'état et de l'adjoint.

Cette première analyse montre que l'algorithme présente une très bonne vitesse de convergence. De plus, la méthode peut donner plusieurs variantes qui présentent différentes propriétés: la méthode des équations de Riccati permet le calcul d'une variation de commande en fonction des perturbations sur l'état mais la stabilité de cette méthode n'est pas garantie, la méthode des sensibilités directes permet une réduction importante du temps de calcul, la méthode de la matrice de transition donne un calcul plus précis des matrices de sensibilités.

Pour améliorer le domaine de convergence relativement limité de cette méthode, nous avons proposé une modification des itérations. La variation des fonctions à chaque itération sera modulée en fonction des erreurs moyennes par rapport à la solution optimale supposée. Par ailleurs, nous avons exploré les différentes possibilités d'initialisation afin d'évaluer la meilleure possible.

Ces développements ont été mis en œuvre dans un programme très général. Toutes les possibilités d'initialisation ainsi que les variantes que nous avons détaillées peuvent être appliqués sur des exemples différents. Les exemples académiques présentés ont permis d'une part de valider le programme du point de vue informatique, d'autre part de confirmer les améliorations proposées pour l'augmentation du domaine de convergence.

Après ces premiers résultats, nous nous proposons d'étudier des applications plus réalistes. En effet, avant d'envisager une application temps réel telle que nous l'avons définie, il est nécessaire de valider l'algorithme sur des exemples plus complexes afin de tester sa robustesse.

CHAPITRE 3

EXEMPLES D'APPLICATION

CHAPITRE 3

EXEMPLES D'APPLICATION EN AERONAUTIQUE

INTRODUCTION

Le but de ce chapitre sera d'une part d'évaluer l'efficacité de la méthode dans des cas plus réalistes, d'autre part d'apporter des compléments de réponse aux problèmes posés par les ingénieurs en matière d'optimisation. Nous nous proposons pour cela d'appliquer la méthode de quasilinearisation modifiée sur deux exemples pris dans le domaine aéronautique.

Le premier cas étudié est celui d'une interception dans le plan horizontal. Nous comparerons sur cet exemple les performances de l'algorithme par rapport à une méthode du premier ordre, en particulier, en fin de convergence. Nous utiliserons pour initialiser les deux méthodes, les résultats donnés par l'approximation des perturbations singulières. La commande obtenue par cette méthode est relativement proche de l'optimale ce qui nous permettra de se placer dans un cas de fin de convergence.

Le second concerne la mécanique du vol hélicoptère. Dans ce domaine, les recherches concernant la modélisation, les commandes de vol électriques, l'optimisation sont d'actualité. En particulier, il existe peu d'études d'optimisation de trajectoires. Le but de cette application est de contribuer à l'étude des performances de vol d'un hélicoptère type. Nous nous proposons de montrer sur un modèle simplifié dans un cas de vol dans le plan vertical, le gain que donne la prise en compte d'une vitesse de rotation du rotor variable. Ces premiers résultats donneront des premiers repères pour de futurs développements.

1. PROBLEME D'INTERCEPTION

1.1. Description du modèle

1.1.1. Definition des repères, du vecteur d'état et du vecteur des commandes.

Les équations d'état décrivant l'évolution du système seront obtenues en utilisant le théorème fondamental de la dynamique. Pour faciliter l'étude des forces appliquées à l'avion, il est classique d'utiliser trois repères de référence (J.C.Wanner 1980). Ces repères ont tous pour origine le centre de gravité de l'avion, les axes étant définis de la façon suivante:

- les axes du repère avion G_x , G_z sont situés dans le plan de symétrie de l'avion, G_x et G_z étant dirigés respectivement vers l'avant et vers le ventre de l'appareil, G_y complétant le repère pour former un trièdre direct

-l'axe G_{x_a} du repère aérodynamique est porté par le vecteur vitesse, G_{z_a} est perpendiculaire à celui-ci dans le plan de symétrie de l'avion, G_{y_a} complète le trièdre direct

-les axes G_{x_0} et G_{y_0} du repère terrestre lié à l'avion définissent le plan horizontal, G_{z_0} étant porté par la verticale descendante.

Les matrices de changements de repère sont donnés en annexe 3.

Nous poserons plusieurs hypothèses avant de mettre en équations le mouvement de l'avion.

H1) L'appareil est assimilé à une masse ponctuelle. Ceci revient à négliger le mouvement de l'attitude de l'avion par rapport à celui de son centre de masse.

H2) Nous supposons que la terre est plate et la pesanteur constante.

H3) La poussée de l'avion et la résultante des forces aérodynamiques sont dans le plan de symétrie de l'avion (vol à dérapage nul).

H4) La masse de l'avion est constante. On néglige en effet la variation de poids due à la consommation de carburant par rapport au poids total de l'avion.

Avec ces hypothèses, l'état du système sera constitué des composantes suivantes $X = (x_G, y_G, h_G, V, \gamma, \chi)$ où x_G, y_G, h_G définissent la position du centre de masse et V, γ, χ sa vitesse.

Les commandes sont au nombre de trois et sont δ_x, μ et α ; δ_x est le taux d'utilisation de la poussée F , μ l'angle de gîte et α l'incidence de l'avion.

La poussée F sera définie par $F = \delta_x F_{\max}(h, M)$ avec $0 \leq \delta_x \leq 1$ (La poussée maximale est fonction de l'altitude et du nombre de mach)

On ne prend pas en compte les constantes de temps ou les transitoires sur les variations de commande.

1.1.2. Equations du mouvement

Faisons le bilan des forces appliquées à l'avion. Ces forces seront repérées dans le trièdre aérodynamique. Il s'agira des forces suivantes

- la résultante des forces aérodynamiques, de composantes R_{xa}, R_{ya}, R_{za}
- la force de propulsion de composantes F_{xa}, F_{ya}, F_{za}
- le poids de l'avion Mg

Les équations cinématiques et l'application du théorème fondamental de la dynamique donnent les relations

$$\frac{d\vec{r}}{dt} = \vec{V}$$

$$\frac{d\vec{V}}{dt} = \Sigma \vec{F}$$

En utilisant les composantes de la vitesse dans le repère aérodynamique, nous obtenons :

$$\begin{aligned} \dot{x} &= V \cos\gamma \cos\chi \\ \dot{y} &= V \cos\gamma \sin\chi \\ \dot{h} &= -V \sin\gamma \\ \dot{V} &= \frac{(F_{xa} + R_{xa})}{M} - g \sin\gamma \\ \dot{\gamma} &= \frac{1}{V} \left(\frac{(R_{za} + F_{za}) \cos\mu}{M} - g \cos\gamma \right) \\ \dot{\chi} &= \frac{1}{MV \cos\gamma} \left((R_{za} + F_{za}) \cdot \sin\mu \right) \end{aligned}$$

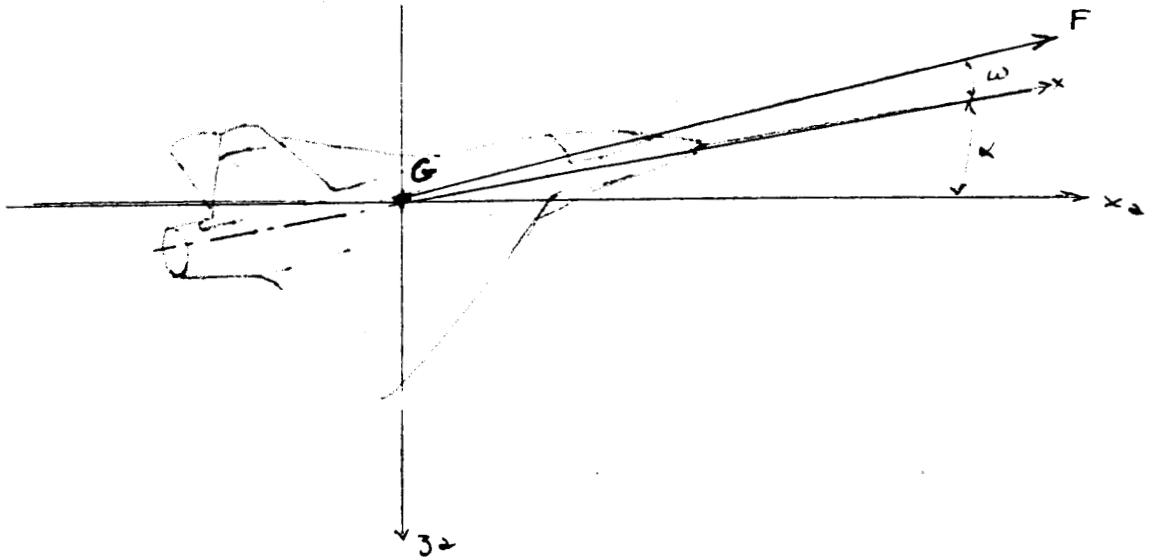
Modèle de propulsion

Nous supposons que la force fait un angle ω avec l'axe de référence Gx de l'avion. Dans le repère aérodynamique, en notant que $\beta = 0$, les composantes F_{xa} et F_{za} seront donc

$$F_{xa} = F \cdot \cos(\alpha + \omega)$$

$$F_{za} = -F \cdot \sin(\alpha + \omega)$$

$$\text{avec } F = \delta_x F_{\max}(h, M)$$



En supposant que l'angle $\alpha + \omega$ est faible, nous négligerons la composante F_{za} par rapport à la portance R_{za} .

Modèle aérodynamique

La résultante des forces aérodynamiques a pour composantes sur le repère aérodynamique:

$$R_{xa} = \frac{1}{2} \rho S V^2 C_x$$

$$R_{za} = \frac{1}{2} \rho S V^2 C_z$$

où ρ est la masse spécifique de l'air, fonction de l'altitude, S la surface de référence de l'avion et C_x, C_z les coefficients de trainée et de portance, fonction de l'incidence et du nombre de Mach.

Les coefficients aérodynamiques seront représentés par une polaire parabolique

$$C_x = C_{x0}(M) + k(M).C_z(M)^2$$

$$C_z = C_{z\alpha}(M).\alpha$$

où C_{x0} , $C_{z\alpha}$ et k sont des fonctions du nombre de Mach

Equations d'état

Considérons le facteur de charge suivant l'axe des z défini par

$$n_z = \frac{(R_{za}+F_{za})}{Mg} \approx \frac{R_{za}}{Mg}$$

avec les expressions de la propulsion et de la force aérodynamique, nous obtenons

$$\frac{(F_{xa}-R_{xa})}{Mg} = \frac{\delta_x F_{max}}{Mg} - D_o - D_1 n_z^2$$

$$n_z = \frac{1}{2mg} \rho S V^2 C_{z\alpha}(M).\alpha$$

$$D_o = \frac{1}{2mg} \rho S V^2 C_{x0}(M)$$

$$D_1 = \frac{2mg}{\rho S V^2} k(M)$$

ce qui donne les équations d'état suivantes

$$\dot{x} = V \cos\gamma \cos\chi$$

$$\dot{y} = V \cos\gamma \sin\chi$$

$$\dot{h} = -V \sin\gamma$$

$$\dot{V} = g \left(\delta_x \frac{F_{max}}{Mg} - D_o - D_1 n_z^2 - \sin\gamma \right)$$

$$\dot{\gamma} = \frac{g}{V} (n_z \cos\mu - \cos\gamma)$$

$$\dot{\chi} = \frac{g n_z \sin\mu}{V \cos\gamma}$$

1.1.3. Définition du domaine de vol

Les commandes et l'état de l'avion seront soumises à différentes contraintes qui seront définies de la façon suivante

1/ contraintes sur la commande,

$$0 \leq \delta_x \leq 1$$

$$n_z \leq \min (n_{z\max}(M,h) , n_{zms})$$

où n_{zms} est une contrainte de structure

et $n_{z\max}$ la limite de décrochage

$$n_{z\max} = \frac{1}{2mg} \rho S V^2 C_{z\alpha}(M) \cdot \alpha_{\max}$$

2/ sur l'état,

$$q = \frac{1}{2} \rho V^2 \leq q_{\max}$$

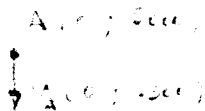
1.2. Formulation générale du problème d'optimisation

1.2.1. Equations du problème.

Nous étudierons un problème d'interception dans le plan horizontal d'un avion de combat avec une cible dont l'évolution est supposée connue. Dans un premier temps, nous supposerons que celle-ci évolue à vitesse constante V_c suivant l'axe des x à altitude constante. Dans un second temps, nous lui donnerons une trajectoire d'évasive à taux de virage constant. Le critère à minimiser sera la distance entre l'avion et la cible à un instant final fixé.

L'état initial de l'avion et de la cible seront figurés ci dessous

z, y



L'hypothèse d'une interception dans un plan horizontal implique:

$$\gamma = \gamma_0 = 0$$

$$n_z = \frac{1}{\cos\mu}$$

Ce qui réduit la dimension de l'état et de la commande. Le système dynamique devient:

$$\begin{aligned} \dot{x} &= V \cos\chi \\ \dot{y} &= V \sin\chi \\ \dot{V} &= \delta_x \frac{F_m}{M} - gD_0 - gD_1 n_z^2 \\ \dot{\chi} &= \frac{g}{V} \sqrt{n_z^2 - 1} \end{aligned}$$

Si nous considérons le vecteur d'état qui représente la position relative de l'avion par rapport à la cible dans le repère terrestre et le vecteur de commandes suivants:

$$\begin{aligned} X &= (x_{\text{avion}} - x_{\text{cible}}, y_{\text{avion}} - y_{\text{cible}}, u_{\text{avion}} - u_{\text{cible}}, v_{\text{avion}} - v_{\text{cible}}) \\ &\text{noté } (x_{ac}, y_{ac}, u_{ac}, v_{ac}) \\ U &= (F_m, n_z) \end{aligned}$$

les équations d'état deviennent (cf annexe 4):

$$\begin{aligned} \dot{x}_{ac} &= u_{ac} \\ \dot{y}_{ac} &= v_{ac} \\ \dot{u}_{ac} &= \frac{g}{V} \left\{ (F_m - (D_0 + D_1 n_z^2)) u_{\text{avion}} - v_{\text{avion}} \sqrt{n_z^2 - 1} \right\} - \dot{u}_{\text{cible}} \\ \dot{v}_{ac} &= \frac{g}{V} \left\{ (F_m - (D_0 + D_1 n_z^2)) v_{\text{avion}} + u_{\text{avion}} \sqrt{n_z^2 - 1} \right\} - \dot{v}_{\text{cible}} \end{aligned}$$

La dynamique de la cible sera donnée par:

- dans le cas d'une évasive rectiligne:

$$\begin{aligned} u_c &= V_c \\ v_c &= 0 \end{aligned}$$

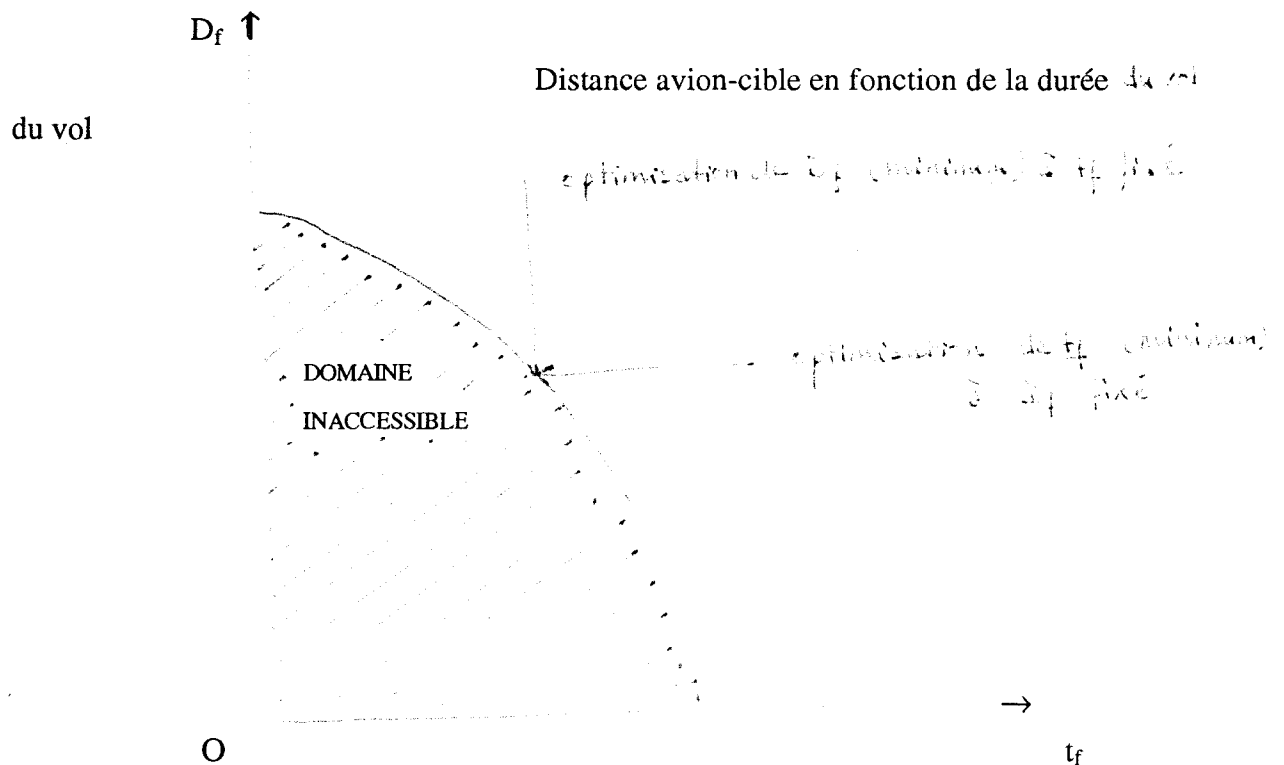
- dans le cas d'une évasive en virage:

$$\begin{aligned} u_c &= V_c \cos(at) \\ v_c &= -V_c \sin(at) \end{aligned}$$

L'indice de performance choisi sera le suivant:

$$J = D_f = \sqrt{x_{ac}(t_f)^2 + y_{ac}(t_f)^2}$$

Il s'agit de minimiser la distance finale entre l'avion et la cible, l'instant final étant fixé. Ce problème est équivalent à celui qui consiste à minimiser le temps nécessaire pour atteindre une distance avion-cible fixée. Il existe en effet une corrélation directe (corrélation biunivoque) entre ces deux problèmes. Ainsi, pour résoudre le problème t_f minimum pour D_f fixé, il suffit de résoudre le problème D_f minimum pour t_f fixé. On en déduit la courbe frontière du domaine accessible (cf fig.), ce qui permettrait d'obtenir la réponse au premier problème (en admettant l'existence d'une seule solution optimale au problème posé).



1.2.2. Résolution analytique

Le Hamiltonien et les équation adjointes associées s'écrivent :

$$\begin{aligned}
 H &= \lambda_x u_{ac} + \lambda_y v_{ac} \\
 &+ \lambda_u \left(\frac{g}{V} \left((F_m - (D_o + D_1 n_z^2)) \cdot u_{avion} - v_{avion} \cdot \sqrt{n_z^2 - 1} \right) - \overset{\circ}{u}_{cible} \right) \\
 &+ \lambda_v \left(\frac{g}{V} \left((F_m - (D_o + D_1 n_z^2)) \cdot v_{avion} + u_{avion} \cdot \sqrt{n_z^2 - 1} \right) - \overset{\circ}{v}_{cible} \right)
 \end{aligned}$$

$$\begin{aligned}
 \dot{\lambda}_x &= 0 & \lambda_x(t_f) &= \frac{x_{acf}}{\sqrt{x_{acf}^2 + y_{acf}^2}} \\
 \dot{\lambda}_y &= 0 & \lambda_y(t_f) &= \frac{y_{acf}}{\sqrt{x_{acf}^2 + y_{acf}^2}} \\
 \dot{\lambda}_u &= -\frac{\partial H}{\partial u_{ac}} & \lambda_u(t_f) &= 0 \\
 \dot{\lambda}_v &= -\frac{\partial H}{\partial v_{ac}} & \lambda_v(t_f) &= 0
 \end{aligned}$$

La solution pour les fonctions adjointes $\lambda_x(t)$ et $\lambda_y(t)$ est immédiate:

$$\begin{aligned}
 \lambda_x(t) &= \frac{x_{acf}}{\sqrt{x_{acf}^2 + y_{acf}^2}} \\
 \lambda_y(t) &= \frac{y_{acf}}{\sqrt{x_{acf}^2 + y_{acf}^2}}
 \end{aligned}$$

Quant aux équations différentielles adjointes pour λ_u et λ_v , elles sont données par:

$$\begin{aligned}
 \dot{\lambda}_u &= -\lambda_x - \lambda_u g \left\{ \frac{\partial K}{\partial V} \frac{u_a^2}{V^2} + \frac{K}{V^3} v_a^2 + \frac{\sqrt{n_z^2 + 1}}{V^3} u_a v_a \right\} \\
 &\quad - \lambda_v g \left\{ \frac{\partial K}{\partial V} \frac{u_a v_a}{V^2} - \frac{K}{V^3} u_a v_a + \frac{\sqrt{n_z^2 + 1}}{V^3} v_a^2 \right\} \\
 \dot{\lambda}_v &= -\lambda_y - \lambda_u g \left\{ \frac{\partial K}{\partial V} \frac{u_a v_a}{V^2} - \frac{K}{V^3} u_a v_a - \frac{\sqrt{n_z^2 + 1}}{V^3} u_a^2 \right\} \\
 &\quad - \lambda_v g \left\{ \frac{\partial K}{\partial V} \frac{v_a^2}{V^2} + \frac{K}{V^3} u_a^2 - \frac{\sqrt{n_z^2 + 1}}{V^3} u_a v_a \right\}
 \end{aligned}$$

avec

$$\begin{aligned}
 V &= \sqrt{u_a^2 + v_a^2} \\
 K &= F_m - D_o - D_1 n_z^2
 \end{aligned}$$

La commande de poussée est une commande de type bang-bang. L'étude sera menée en prenant une commande de poussée maximale, ce qui est le cas d'une interception normale. La commande en facteur de charge sera donnée par

$$n_z^* = \sqrt{\left(\frac{\lambda_v u_a - \lambda_u v_a}{2D_1 (\lambda_u u_a + \lambda_v v_a)} \right)^2 + 1}$$

1.2.3. Résultats numériques

Les premiers essais d'optimisation ont confirmé la très grande sensibilité de l'algorithme aux erreurs sur les fonctions initiales. Comme nous l'avons remarqué, dans le chapitre précédent, l'utilisation de la méthode de quasilinearisation est intéressante surtout en fin de convergence, lorsque les fonctions se rapprochent de l'optimum et que les méthodes du premier ordre ne permettent plus d'obtenir une amélioration en peu d'itérations.

Initialisation

La technique des perturbations singulière nous permet d'initialiser l'algorithme par une solution quasioptimale. La commande nominale est alors donnée par (cf. annexe 5)

$$n_{z\text{pert}} = \frac{1}{\cos\mu_{\text{pert}}}$$
$$\tan\mu_{\text{pert}} = \sqrt{\frac{2V}{\bar{V} - V}} \cdot \sin\frac{(\bar{\chi} - \chi)}{2} \cdot \frac{F_m - D_o - D_1}{D_1}$$
$$F_m(\bar{V}) - D_o(\bar{V}) - D_1(\bar{V}) = 0$$
$$V_c \sin\bar{\psi} = \bar{V} \sin(\bar{\psi} - \bar{\chi})$$
$$\tan\bar{\psi} = \frac{y_c - y_a}{x_c - x_a}$$

Nous avons traité le problème dans les deux cas de figure pour deux altitudes différentes:

$$h_1 = 2000 \text{ m}$$

$$h_2 = 9000 \text{ m}$$

Les essais ont été réalisés pour $t_f = 12 \text{ s}$, 60 s et 80 s . Les résultats sont donnés au tableau page suivante.

Exemple			Distance avion-cible (km)				Nombre d'itérations avant convergence	
t_f	altitude	évasive	D_o^0	D_f^0	D_f^{GP}	D_f^{QSL}	G P	QSL
12 s	2000 m	rectiligne	25,49	25,24	23,36	23,3	11	3
		en virage	-	25,25	23,37	23,36	11	5
	9000 m	rectiligne	-	25,38	25,12	25,12	2	1
		en virage	-	25,39	25,19	25,20	1	1
60 s	2000 m	rectiligne	25,49	16,87	15,03	15,02	> 50	10
		en virage	-	16,91	14,76	14,79	> 50	8
	9000 m	rectiligne	25,49	19,03	18,50	18,502	18	2
		en virage	-	18,87	18,01	18,01	22	5
80 s	2000 m	rectiligne	25,49	11,87	10,20	10,21	> 50	21
		en virage	-	11,90	9,46	9,50	> 50	15
	9000 m	rectiligne	25,49	14,77	13,84	13,85	35	14
		en virage	-	14,42	12,66	12,70	31	11

Commentaires:

Ces résultats montrent la grande vitesse de convergence de l'algorithme. Cependant pour compléter la comparaison, il est nécessaire d'évaluer le nombre d'opérations nécessaires à chaque itération pour les deux algorithmes.

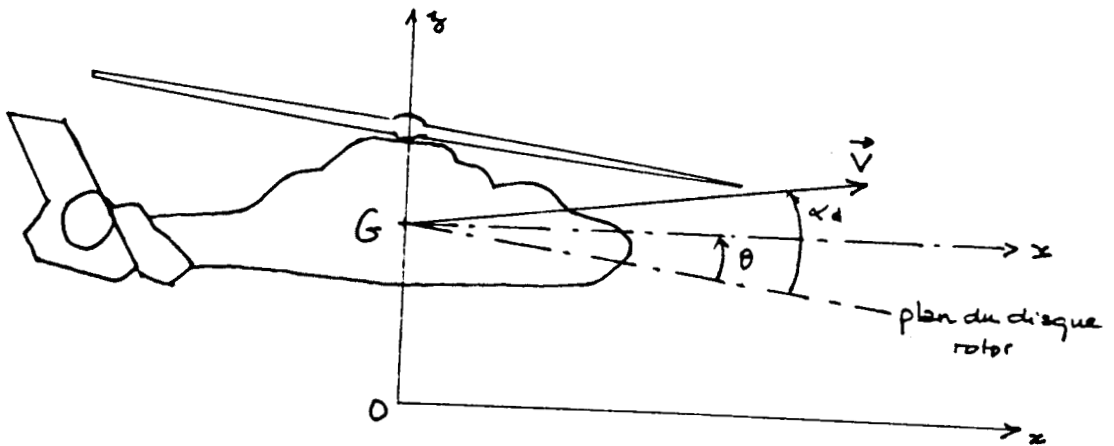
A ce propos, nous signalons que les résultats donnés par la quasilinearisation ont été obtenus avec un degré d'analyticit  relativement important.

2. EVALUATION DES PERFORMANCES D'UN HELICOPTERE POUR UNE MANOEUVRE D'EVITEMENT D'OBSTACLE.

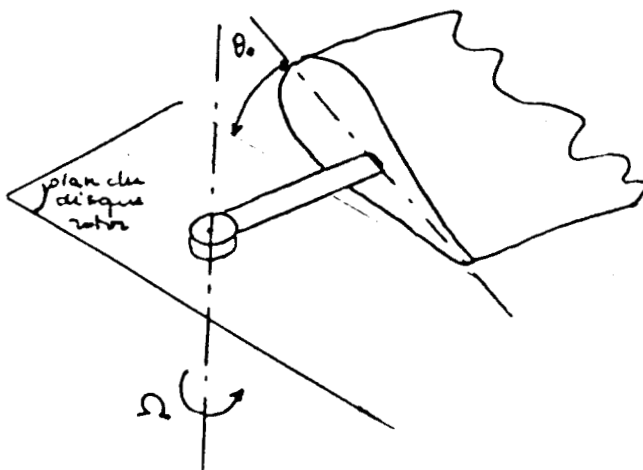
2.1. Description du modèle physique

2.1.1. Définition des repères, du vecteur d'état et de commandes

Nous présentons ici un modèle hélicoptère de type point matériel. Nous définissons tout d'abord les repères et les angles qui serviront à décrire la dynamique du système.



(Ox, y, z) repère terrestre
 θ assiette du disque rotor
 α_d incidence du disque rotor
 V vitesse de l'appareil



θ_0 pas collectif

Le modèle décrit la dynamique de deux composantes de l'appareil: le centre de gravité et le rotor.

- Le mouvement du centre de gravité est décrit par les équations de la dynamique du point matériel. Les forces sont ramenées au centre de gravité et sont dans une première approximation, la portance de rotor, le poids de l'hélicoptère et la trainée du fuselage. Nous négligerons la portance du fuselage beaucoup plus faible que celle du rotor principal.

- L'équation de régime du rotor, en rotation sur un axe relié au moteur par l'intermédiaire de la boîte de transmission principale, traduit une conservation d'énergie entre la puissance fournie au rotor par le moteur, la puissance dissipée par le rotor et la variation d'énergie cinétique de rotation du rotor.

Les mouvements de ces deux composantes sont couplés par la portance du rotor et les vitesses de déplacement.

Nous avons négligé dans ce modèle, le mouvement du fuselage en rotation autour de son centre de gravité. Ceci suppose que le fuselage est constamment équilibré en rotation. Cette hypothèse est réaliste car on calcule le modèle de puissance nécessaire au rotor à partir des conditions de vol équilibré. Par ailleurs, nous avons négligé la dynamique de l'assiette du fuselage et du disque rotor, plus rapide que celle du centre de gravité (hypothèse des échelles de temps séparables).

Etat de l'hélicoptère

L'état est composé des six variables qui décrivent le mouvement du centre de gravité et de la variable régime du rotor. Soit

X	position longitudinale du fuselage dans le repère terrestre
V_X	vitesse longitudinale
Y	position latérale du fuselage
V_Y	vitesse latérale
H	altitude du fuselage
V_Z	vitesse verticale
Ω	régime rotor

Commandes

Les commandes seront au nombre de quatre. Soit

- θ_o pas collectif des pales du rotor principal (cf fig.)
- P_m puissance fournie par le moteur
- α_d incidence du disque rotor
- ϕ assiette latérale du disque rotor

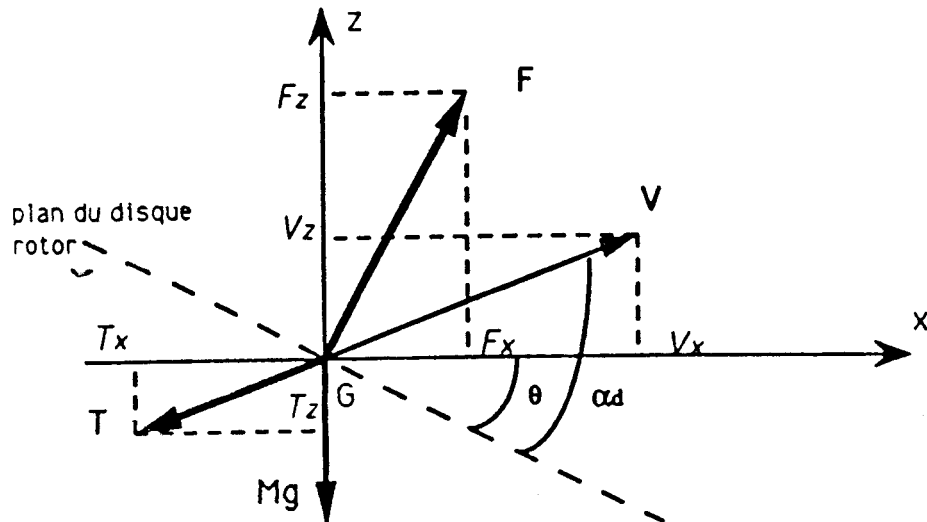
3.1.2. équations du mouvement dans le plan vertical

Nous considérerons le mouvement de l'hélicoptère dans un plan vertical. L'état sera alors de dimension quatre (composantes latérales Y et Vy nulles) et la commande de dimension trois (commande en latéral nulle).

Dynamique du centre de gravité :

Les forces appliquées au centre de gravité de l'hélicoptère sont au nombre de trois (cf fig.) :

- \vec{F} la portance du rotor, perpendiculaire au plan du disque rotor
- \vec{Mg} le poids de l'hélicoptère
- \vec{T} la traînée du fuselage en sens opposé à la vitesse.



V vitesse de l'appareil

α_d incidence du disque rotor

θ assiette du disque rotor

$\alpha_d < 0$ dans ce cas de figure

$\theta < 0$ dans ce cas de figure

Modèle aérodynamique

Le module de la portance rotor sera notée F_n . Pour tenir compte de l'effet de la déportance du fuselage, la composante verticale F_z sera corrigée par un facteur correctif $D_{\text{éport}}$ qui dépend de la vitesse horizontale V_x . Ce qui donne

$$\begin{aligned} F_x &= -F_n \sin\theta && \text{positive vers l'avant} \\ F_z &= \frac{F_n \cos\theta}{D_{\text{éport}}} && \text{positive vers le haut} \end{aligned}$$

La trainée du fuselage sera donnée par $\frac{1}{2} \rho \cdot (SC_x)_{\text{fuselage}} V^2$ soit

$$\begin{aligned} T_x &= -\frac{1}{2} \rho \cdot (SC_x)_f V V_x \\ T_z &= -\frac{1}{2} \rho \cdot (SC_x)_f V V_z \end{aligned}$$

avec

$$\begin{aligned} V &= \sqrt{V_x^2 + V_z^2} \\ V_x &= V \cos(\alpha_d - \theta) \\ V_z &= -V \sin(\alpha_d - \theta) \end{aligned}$$

L'application du théorème de la dynamique donne alors

$$\begin{aligned} \dot{x} &= V_x \\ \dot{h} &= V_z \\ MV_x^\circ &= -F_n \sin\theta - \frac{1}{2} \rho \cdot (SC_x)_f V V_x \\ MV_z^\circ &= \frac{F_n \cos\theta}{D_{\text{éport}}} - \frac{1}{2} \rho \cdot (SC_x)_f V V_z - Mg \end{aligned}$$

avec

$$\begin{aligned} \sin\theta &= \frac{V_z \cos\alpha_d + V_x \sin\alpha_d}{V} \\ \cos\theta &= \frac{V_x \cos\alpha_d - V_z \sin\alpha_d}{V} \end{aligned}$$

La portance du disque rotor est donnée par

$$F_n = \frac{1}{6} \rho S \sigma (\Omega.R)^2 C_{zm}$$

où C_{zm} est le coefficient moyen de portance du rotor.

Le calcul du coefficient C_{zm} est assez complexe. En effet, il résulte du pas collectif et de la vitesse de l'écoulement de l'air au niveau des pales du rotor principal. Dans cette vitesse interviennent

- la vitesse de l'appareil
- la vitesse induite V_i calculée à partir de la théorie de Froude.

Or ces vitesses dépendent elles même de la portance F_n et donc de C_{zm} .

Nous sommes donc amenés à résoudre à chaque instant une équation implicite relativement complexe.

Si nous définissons les paramètres de flux d'avancement : μ_d dans le plan du disque rotor et λ_z perpendiculaire au plan du disque rotor

$$\mu_d = \frac{V \cdot \cos \alpha_d}{\Omega \cdot R}$$

$$\lambda_z = \frac{V \cdot \sin \alpha_d}{\Omega \cdot R}$$

nous aurons $C_{zm} = KC_{zmo} + KC_{zm1} \cdot \lambda_i$

avec $KC_{zmo} = 3 C_{z\alpha} \left(\frac{1}{3} + \frac{\mu_d^2}{2} - \frac{\frac{4}{3} \mu_d^2}{1 + \frac{3}{2} \mu_d^2} \right) \theta_0 - KC_{zm1} \cdot \lambda_z$

$$KC_{zm1} = -\frac{3}{2} C_{z\alpha} \left(1 - \frac{2\mu_d^2}{1 + \frac{3}{2} \mu_d^2} \right)$$

$$\lambda_i = \frac{\frac{\sigma}{12}}{\sqrt{\mu_d^2 + (\lambda_i - \lambda_z)^2}} \cdot C_{zm}$$

L'équation implicite à résoudre sera alors

$$\lambda_i = \frac{\frac{\sigma}{12}}{\sqrt{\mu_d^2 + (\lambda_i - \lambda_z)^2}} \cdot (KC_{zmo} + KC_{zm1} \cdot \lambda_i)$$

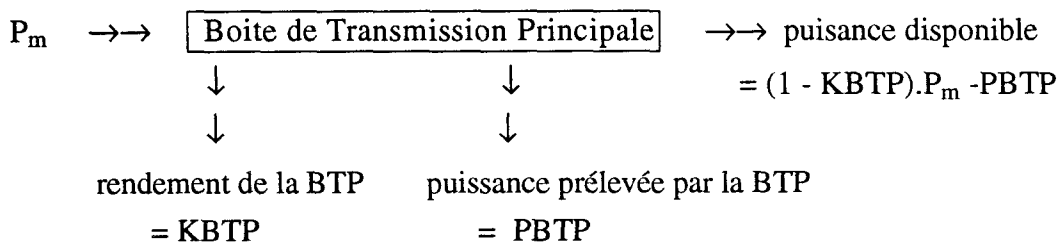
bilan de puissance

L'équation du moment dynamique du rotor principal donne

$$I_x \Omega \dot{\Omega} = P_{\text{disp}} - P_{\text{nec}}$$

Calcul de la puissance disponible sur le rotor:

Cette puissance est liée à la puissance P_m fournie par le moteur à la boîte de transmission suivant le schéma suivant



La puissance disponible se répartit en une puissance P_{rp} sur le rotor principal et une puissance P_{ra} sur le rotor arrière. Cette dernière étant proportionnelle à P_{rp} par un coefficient KRA, nous aurons le bilan suivant

$$P_{\text{disp}} = \frac{(1 - KBTP).P_m - PBTP}{(1 + KRA)}$$

Calcul de la puissance nécessaire:

La puissance nécessaire au rotor principal en vol est la somme de plusieurs puissances: la puissance induite, la puissance de profil et la puissance de déplacement. La puissance induite est donnée par:

$$P_{\text{ind}} = \frac{K}{B} F_n V_{\text{ind}}$$

où $\frac{K}{B}$ est le coefficient de portance induite. La puissance de profil nécessaire pour vaincre la traînée de profil des pales est donnée par:

$$P_{\text{prof}} = \frac{1}{8} \rho S \sigma (\Omega R)^3 C_{xp}$$

où C_{xp} est un coefficient qui dépend de C_{zm} et de μ_d . La puissance de déplacement est donnée par:

$$P_{\text{depl}} = (-F_n \sin\theta).V_x + (F_n \cos\theta).V_z$$

Les équations d'état seront donc:

$$\begin{aligned}\dot{x} &= V_x \\ \dot{h} &= V_z \\ \dot{V}_x &= \frac{1}{M} \left(-F_n \sin\theta - \frac{1}{2} \rho \cdot (SC_x)_f V V_x \right) \\ \dot{V}_z &= \frac{1}{M} \left(F_n \cos\theta - \frac{1}{2} \rho \cdot (SC_x)_f V V_z \right) - g \\ \dot{\Omega} &= \frac{P_{disp} - P_{nec}}{I_x \Omega}\end{aligned}$$

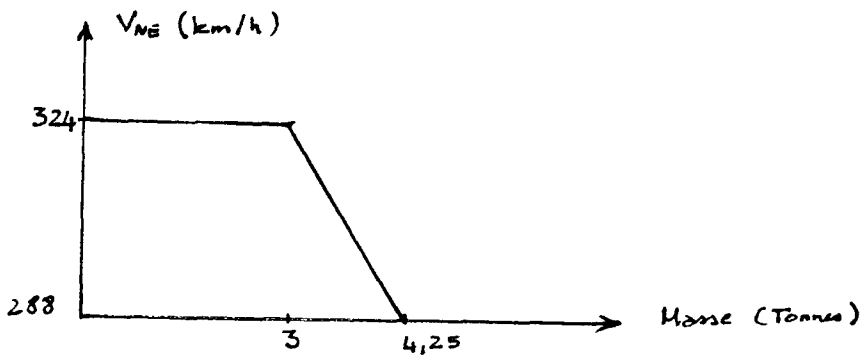
avec
$$P_{disp} = \frac{(1 - KBTP) \cdot P_m - PBTP}{(1 + KRA)}$$

2.1.3. Définition du domaine de vol

Le domaine de vol de l'hélicoptère sera limité par des contraintes sur l'état et/ou sur la commande.

1/ Les contraintes sur la commande sont les bornes imposées au pas collectif (contrainte de décrochage du rotor), à la puissance motrice (puissance maximale fournie par le moteur) et aux commandes d'assiette (contraintes d'équilibrage).

2/ Les contraintes sur l'état portent sur la vitesse et l'altitude du centre de gravité. Les limites de capacité de l'appareil (limites du moteur et limites de structure) donnent un vitesse à ne pas excéder (cf. fig). Nous imposerons comme borne supérieure la vitesse nominale opérationnelle maximale égale à 90% de la vitesse V_{NE} .



$$H \geq H_{min}$$

$$V \leq V_{max} \quad V_{max} = 0,9 \cdot V_{NE}$$

3/ Les contraintes sur l'état et la commande viennent de la marge de controllabilité de l'hélicoptère. Lorsque le moment exercé par le rotor sur le fuselage diminue, la controllabilité de l'appareil diminue et peut même s'annuler si le facteur de charge devient négatif. On ne peut plus commander ou stabiliser l'assiette du fuselage. Or ce moment dépend de la portance rotor, on imposera donc une borne inférieure à la portance rotor.

2.1.4. Simplification du modèle

Traitement de la contrainte sur l'état

La contrainte sur le régime moteur sera transformée en une contrainte sur la commande. Nous remplaçons la commande de puissance motrice par un régime rotor désiré. L'équation de régime sera alors donnée par un premier ordre par rapport à ce régime désiré soit: $\dot{\Omega} = \frac{\Omega_{\text{dés}} - \Omega}{\tau}$

Simplification des équations d'état

Dans le cas de manœuvre considéré, l'altitude est de l'ordre de 500 mètres. Nous considérerons que ρ reste constant.

Par ailleurs nous négligerons la déportance due au fuselage, l'erreur étant inférieure à 2%. Cela revient à poser:

$$\text{Déport} \approx 1$$

Pour le calcul de la portance, nous supposons que

$$(\lambda_z - \lambda_i)^2 \ll \mu_d^2$$

Dans ce cas, il n'y a plus d'équation implicite à résoudre et nous avons (μ_d étant toujours positif)

$$\lambda_i \approx \frac{\sigma \cdot K C_{zmo}}{12 \cdot \mu_d} \cdot \frac{1}{1 - \frac{\sigma \cdot K C_{zml}}{12 \cdot \mu_d}}$$

$$C_{zm} \approx \frac{KC_{zmo}}{1 - \frac{\sigma \cdot KC_{zm1}}{12 \cdot \mu_d}}$$

Ces hypothèses seront vérifiées à posteriori.

Contraintes sur les commandes

Le pas collectif ainsi que l'incidence du disque rotor sont limités par les bornes suivantes:

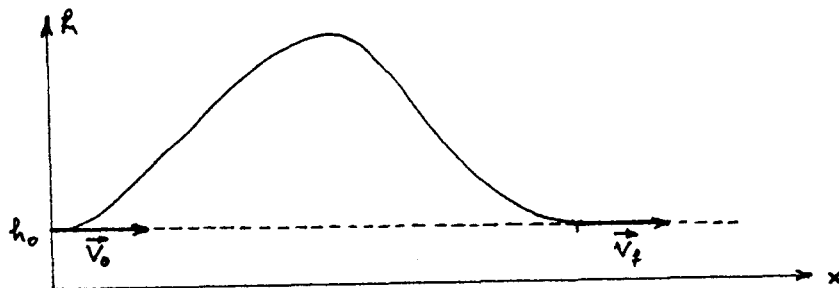
$$\begin{aligned} 0^\circ \leq \theta_o \leq 8^\circ \\ -20^\circ \leq \alpha_d \leq 20^\circ \end{aligned}$$

Ces bornes sont plus sévères par rapport à celles imposées habituellement sur tout le domaine de vol ($\theta_{o\max}=10,5^\circ$ et $|\alpha_d| \leq 30^\circ$). En fait, celles-ci correspondent à un coefficient de portance réaliste dans notre cas de manœuvre, étant donné le modèle simplifié que nous avons considéré.

2.2. Evaluation du gain en performance par le contrôle actif moteur

2.2.1. Problème d'optimisation

Considérons la manœuvre suivante qui consiste à maximiser l'altitude sur une distance donnée avec des vitesses initiale et finale horizontales et une altitude finale égale à l'altitude initiale. (cf fig.).



Nous choisirons le critère suivant

$$\text{maximiser la surface } S = \int_{t_0}^{t_f} (h - h_0) \cdot dx$$

avec les conditions initiales et finales

$$\begin{array}{ll} x(t_0) = 0 & x(t_f) \text{ libre} \\ h(t_0) = h_0 & h(t_f) = h_0 \\ V_x(t_0) = V_{x0} & V_x(t_f) \text{ libre} \\ V_z(t_0) = 0 & V_z(t_f) = 0 \\ \Omega(t_0) = \Omega_0 & \Omega(t_f) \text{ libre} \end{array}$$

Ω_0 correspond au régime nominal du moteur

Pour éviter la commande en tout ou rien sur θ_0 et sur Ω_d , nous définirons le critère suivant à minimiser:

$$J = - \int_{t_0}^{t_f} (h - h_0) \cdot V_x dt + K \cdot \int_{t_0}^{t_f} \left(\frac{\Omega_d - \Omega_0}{\Omega_0} \right)^2 + \left(\frac{\theta_0 - \theta_{00}}{\theta_{00}} \right)^2 dt$$

2.2.2. Résolution analytique

L'Hamiltonien et les équations adjointes seront donnés par:

$$\begin{aligned} H = & - (h - h_0) V_x + K \cdot \left(\frac{\Omega_d - \Omega_0}{\Omega_0} \right)^2 + K \cdot \left(\frac{\theta_0 - \theta_{00}}{\theta_{00}} \right)^2 \\ & + \lambda_x V_x + \lambda_h V_z \\ & + \lambda_{vx} \left(\frac{1}{M} \left(-F_n \sin \theta - \frac{1}{2} \rho \cdot (SC_x)_f V V_x \right) \right) \\ & + \lambda_{vz} \left(\frac{1}{M} \left(F_n \cos \theta - \frac{1}{2} \rho \cdot (SC_x)_f V V_z \right) - g \right) \\ & + \lambda_{\Omega} \left(\frac{\Omega_{dés} - \Omega}{\tau} \right) \end{aligned}$$

$$\begin{array}{ll} \dot{\lambda}_x = 0 & \lambda_{xf} = 0 \\ \dot{\lambda}_h = V_x & \lambda_{hf} = v_z \\ \dot{\lambda}_{vx} = - \frac{\partial H}{\partial V_x} & \lambda_{vxf} = 0 \\ \dot{\lambda}_{vz} = - \frac{\partial H}{\partial V_z} & \lambda_{vzf} = v_{vz} \\ \dot{\lambda}_{\Omega} = - \frac{\partial H}{\partial \Omega} & \lambda_{\Omega f} = 0 \end{array}$$

Le calcul de la commande optimale sera simple pour Ω_d et θ_o .

$$\frac{\partial H}{\partial \Omega_d} = \frac{2K}{\Omega_o^2} (\Omega_d - \Omega_o) + \frac{\lambda_\Omega}{\tau}$$

$$\frac{\partial H}{\partial \theta_o} = \frac{2K}{\theta_{oo}^2} (\theta_o - \theta_{oo}) + \frac{\partial F_n}{\partial \theta_o} \cdot \frac{(-\lambda_{vx} \sin \theta + \lambda_{vz} \cos \theta)}{M}$$

$$\text{avec } \frac{\partial F_n}{\partial \theta_o} = \frac{1}{6} \rho S \sigma (\Omega R)^2 \cdot 3 \cdot C_{z\alpha} \frac{\left(\frac{1}{3} + \frac{\mu_d^2}{2} - \frac{\frac{4}{3} \mu_d^2}{1 + \frac{3}{2} \mu_d^2} \right)}{\left(1 - \frac{\sigma \cdot K C_{zm1}}{12 \cdot \mu_d} \right)}$$

ce qui donne

$$\Omega_d^* = - \frac{\lambda_\Omega \Omega_o^2}{\tau 2K} + \Omega_o$$

$$\theta_o^* = - \frac{\partial F_n}{\partial \theta_o} \cdot \frac{-\lambda_{vx} \sin \theta + \lambda_{vz} \cos \theta}{M} \cdot \frac{\theta_{oo}^2}{2K} + \theta_{oo}$$

L'incidence optimale du disque rotor sera donnée par une équation implicite

$$\frac{\partial H}{\partial \alpha_d} = 0 \quad \text{avec}$$

$$\frac{\partial H}{\partial \alpha_d} = \frac{1}{M} \frac{\partial F_n}{\partial \alpha_d} (-\lambda_{vx} \sin \theta + \lambda_{vz} \cos \theta) + \frac{1}{M} F_n (-\lambda_{vx} \cos \theta - \lambda_{vz} \sin \theta)$$

avec

$$\frac{\partial F_n}{\partial \alpha_d} = \frac{1}{6} \rho S \sigma (\Omega R)^2 \cdot \frac{1}{\left(1 - \frac{\sigma \cdot K C_{zm1}}{12 \cdot \mu_d} \right)^2} \cdot G$$

$$G = - \mu_d \left(1 - \frac{\sigma \cdot K C_{zm1}}{12 \cdot \mu_d} \right) \left(3 \cdot C_{z\alpha} \cdot \theta_o \cdot \lambda_z \cdot \left(1 - \frac{\frac{8}{3} \mu_d}{\left(1 + \frac{3}{2} \mu_d^2 \right)^2} \right) \right.$$

$$\left. - 6 \cdot C_{z\alpha} \cdot \lambda_z^2 \cdot \left(\frac{\mu_d}{\left(1 + \frac{3}{2} \mu_d^2 \right)^2} \right) + K C_{zm1} \right)$$

$$- \lambda_z K C_{zmo} \frac{\sigma}{12} \left(6 \cdot C_{z\alpha} \cdot \frac{1}{\left(1 + \frac{3}{2} \mu_d^2 \right)^2} - \frac{K C_{zm1}}{\mu_d^2} \right)$$

Analyse de l'incidence optimale :

D'après leur définition, nous remarquons que μ_d et λ_z sont des valeurs relativement petites devant l'unité si la vitesse n'est pas trop grande, ce qui est le cas pour notre type de manœuvre. En effet, pour le régime nominal et une vitesse moyenne de 20 m/s nous avons $\mu_d < 1/10 \cos \alpha_d$ et $\lambda_z < 1/10 \sin \alpha_d$

Dans ce cas, nous avons les approximations suivantes

$$\begin{aligned}
 KC_{zm1} &\approx -\frac{3}{2} C_{z\alpha} \\
 KC_{zmo} &\approx C_{z\alpha} \left(\theta_o + \frac{3}{2} \lambda_z \right) \\
 C_{zm} &\approx \frac{C_{z\alpha} \left(\theta_o + \frac{3}{2} \lambda_z \right)}{1 + \frac{\sigma \cdot C_{z\alpha}}{8 \cdot \mu_d}} \\
 G &\approx C_{z\alpha} \left(\frac{3}{2} \cdot \mu_d \cdot \left(1 + \frac{\sigma \cdot C_{z\alpha}}{8 \cdot \mu_d} \right) - \frac{\sigma \cdot C_{z\alpha}}{8 \cdot \mu_d^2} \lambda_z \cdot \left(\theta_o + \frac{3}{2} \lambda_z \right) \right)
 \end{aligned}$$

et l'équation implicite devient

$$\frac{G}{C_{z\alpha} \left(\theta_o + \frac{3}{2} \lambda_z \right) \left(1 + \frac{\sigma \cdot C_{z\alpha}}{8 \cdot \mu_d} \right)} = \frac{(\lambda_{vx} \cos \theta + \lambda_{vz} \sin \theta)}{(-\lambda_{vx} \sin \theta + \lambda_{vz} \cos \theta)}$$

soit

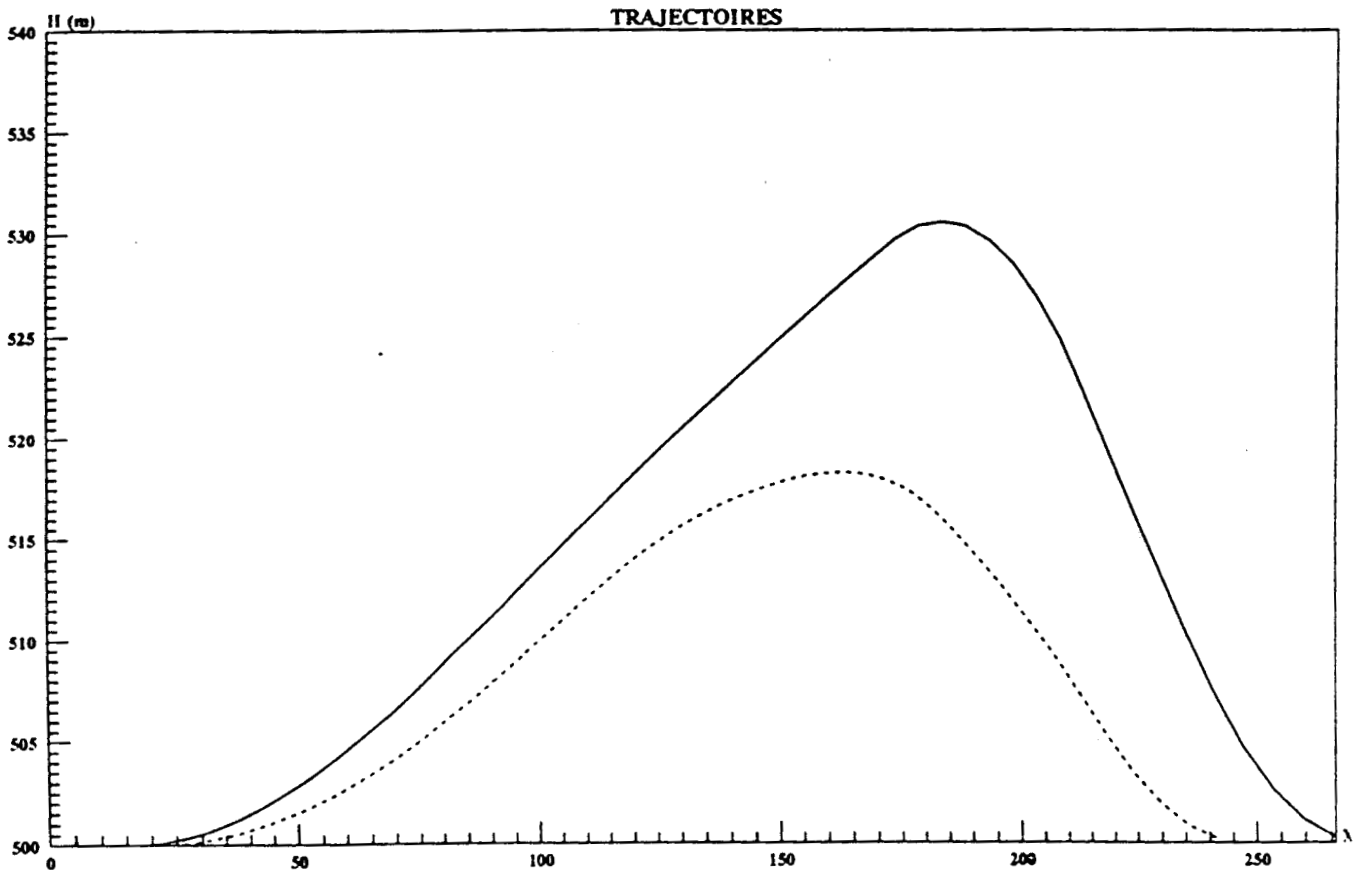
$$\frac{\frac{3}{2} \cdot \mu_d}{\left(\theta_o + \frac{3}{2} \lambda_z \right)} - \frac{\frac{\lambda_z \sigma \cdot C_{z\alpha}}{\mu_d \cdot 8 \cdot \mu_d}}{\left(1 + \frac{\sigma \cdot C_{z\alpha}}{8 \cdot \mu_d} \right)} = \frac{(\lambda_{vx} \cos \theta + \lambda_{vz} \sin \theta)}{(-\lambda_{vx} \sin \theta + \lambda_{vz} \cos \theta)}$$

2.2.3. Résultats numériques

Afin d'évaluer l'amélioration des capacités de l'appareil par l'utilisation d'une vitesse de rotation variable du rotor, deux problèmes seront traités. Dans le premier cas la vitesse de rotation restera fixée, dans le second cas, nous permettrons une variation de 10% par rapport à la vitesse de rotation rotor nominale.

Des premiers calculs seront effectués par le gradient projeté afin d'approcher la solution optimale. L'initialisation de l'algorithme est donnée par les commandes de vol horizontal équilibré, la vitesse horizontale étant fixée égale à la vitesse initiale. Le calcul est donné en annexe 6.

Pour l'optimisation du système sans contrôle actif moteur, la vitesse désirée de rotation du rotor est fixée à 1% de la valeur nominale (36,6 rad/s). Les deux trajectoires obtenues ainsi que les commandes correspondantes sont figurées ci dessous. Les courbes obtenues sont figurées planches 11 à 14 et 19 à 22. Nous avons présenté les commandes, les variables d'état, les adjoints et certaines variables auxiliaires qui nous ont permis de vérifier à posteriori les simplifications adoptées.



CONCLUSION

Nous avons appliqué la quasilinéarisation à deux exemples le plus réalistes possibles du domaine aéronautique.

Le premier exemple montre que pour un problème d'optimisation sans contraintes finales la quasilinéarisation permet d'obtenir la commande optimale en un nombre réduit d'itérations. Comme nous l'avons signalé, bien que le nombre d'opérations effectuées à chaque itération est beaucoup plus important que pour le gradient projeté, le gain en temps de calcul reste intéressant du fait de la réduction significative en nombre d'itérations.

Le second exemple visait à confirmer l'intérêt d'utilisation du contrôle de la rotation du rotor principal d'un hélicoptère pour augmenter les capacité de l'appareil. Dans le cas de manœuvre que nous avons considéré, les performances sont nettement améliorées. L'utilisation de la quasilinéarisation permet de confirmer rapidement les premiers résultats obtenus par le gradient projeté.

Nous signalons que la quasilinéarisation demande une étude analytique relativement importante pour le calcul analytique de la commande. Cette étude ne sera pas toujours possible pour des systèmes très complexes. Cependant, dans le cadre de l'application temps réel que nous avons définie, cette réserve peut être minimisée. En effet, le modèle utilisé pour l'optimisation peut être simplifié de façon à permettre une étude analytique, la réactualisation de la commande devra alors être effectuée à des intervalles relativement proches à condition que les perturbations ou les erreurs restent relativement faibles.

CONCLUSION GENERALE

Nous avons effectué une synthèse des différentes méthodes de calcul de commande optimale développées dans la littérature. Cette synthèse destinée à tous ceux qui cherchent pour un cas et pour une utilisation donnés, la méthode la plus appropriée, montre la grande diversité des approches développées à ce jour. Parmi ces méthodes, nous avons distingué trois groupes dont nous donnons les caractéristiques générales:

- Les méthodes qui permettent de calculer directement la commande optimale et ont un champ d'application relativement restreint. Celles ci ont l'avantage de donner une expression analytique de la solution. De plus, la commande obtenue est souvent une commande en boucle fermée.

- Les méthodes paramétriques qui peuvent s'adapter à tous les types de problèmes. Ces méthodes donnent une approximation de la trajectoire optimale et le type de paramétrisation sera particulier à chaque application.

- Les méthodes numériques générales qui souffrent soit d'une faible vitesse de convergence, soit d'une grande sensibilité à l'initialisation. De ce fait, pour résoudre un problème de commande optimale général, il faudra combiner plusieurs méthodes numériques.

Parmi celles-ci, il a été intéressant, dans le cadre des études menées l'O.N.E.R.A., d'étudier une méthode indirecte à grande vitesse de convergence pour compléter une méthode de gradient projeté. En effet, la convergence quadratique de la quasilinearisation et la possibilité d'atteindre par cet algorithme un optimum local nous a amené à analyser cette méthode de manière plus précise.

Cette analyse laisse espérer une très bonne vitesse de convergence en phase terminale et la possibilité de traiter un certain nombre de contraintes, mais a souligné une faiblesse, en l'occurrence, la grande sensibilité à l'initialisation. Afin de remédier à cette faiblesse, nous avons développé une gestion particulière du pas de l'algorithme qui a permis d'élargir le domaine de convergence. Ce calcul tient compte de l'erreur de la trajectoire nominale par rapport à une trajectoire optimale estimée.

Nous avons élaboré un programme relativement général pour mettre en œuvre ces développements. Ce programme est modulaire, permettant des développements ultérieurs aisés. Testé sur des exemples simples, il nous a permis de montrer les possibilités de l'algorithme modifié. Bien que la gestion des pondérations reste dans la pratique un problème relativement difficile, l'algorithme a donné de bons résultats d'une part sur un cas d'interception plan d'un avion de combat, d'autre part sur un cas de manœuvre d'un hélicoptère.

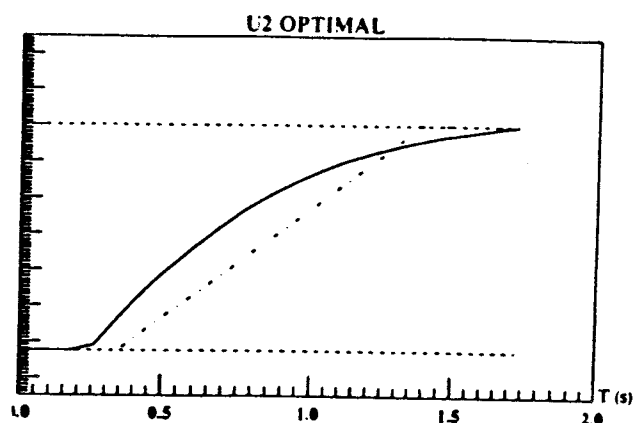
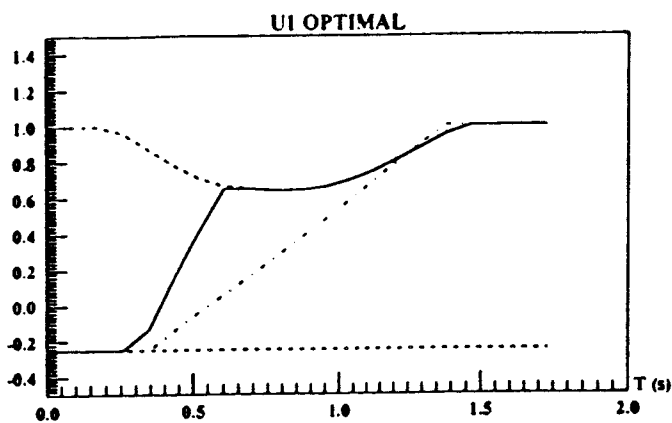
Ces exemples réalistes confirment que pour une initialisation proche de la solution, l'algorithme donne un très bon résultat en peu d'itérations. De plus, nous avons complété certains résultats obtenus par une méthode du premier ordre, le gradient projeté, sur l'intérêt d'une vitesse de rotation de rotor variable pour un hélicoptère, problème pour lequel peu de résultats existent actuellement.

Ce programme pourra donc servir d'outil aux ingénieurs pour résoudre un problème d'optimisation ou compléter des résultats obtenus par d'autres méthodes.

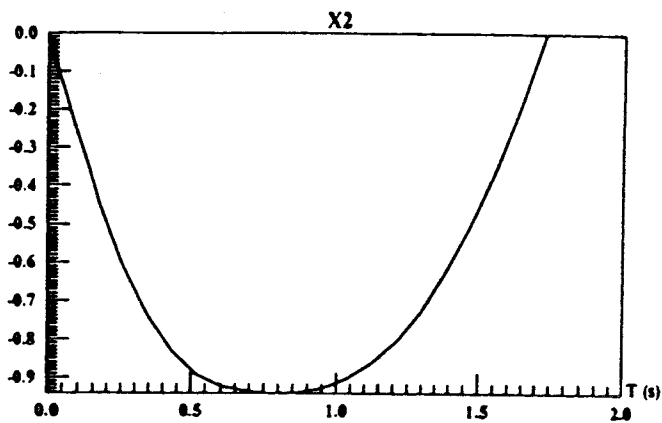
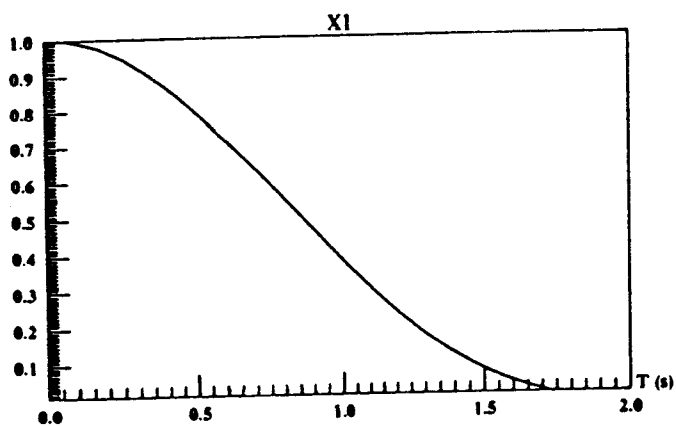
Dans l'avenir, ce programme pourra être développé en y ajoutant d'autres modules ce qui permettra d'augmenter sa fiabilité, de disposer d'un large éventail de méthodes et d'exploiter la quasilinéarisation dans différents contextes.

En particulier, ses qualités de convergence pourraient être exploitées dans l'évaluation d'un guidage à partir d'une trajectoire optimale précalculée. En effet, il peut être envisagé de recalculer à certains instants une nouvelle trajectoire optimale en tenant compte de l'écart par rapport à la trajectoire optimale nominale, écart dû à des perturbations ou à des erreurs éventuelles de modélisation.

PLANCHES

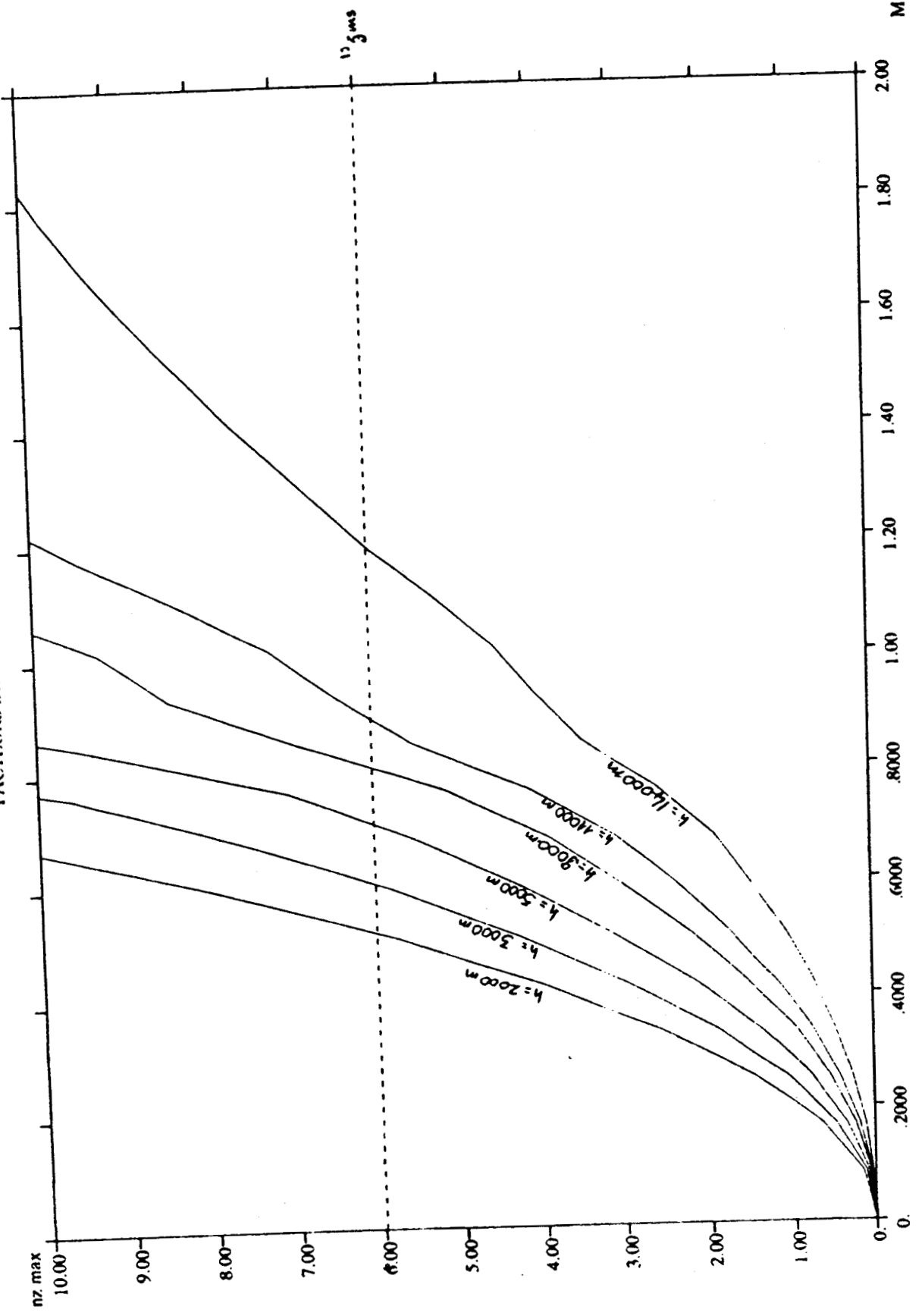


- - - - - initialisation
 ——— commande obtenue
 - - - - - bornes sur la commande

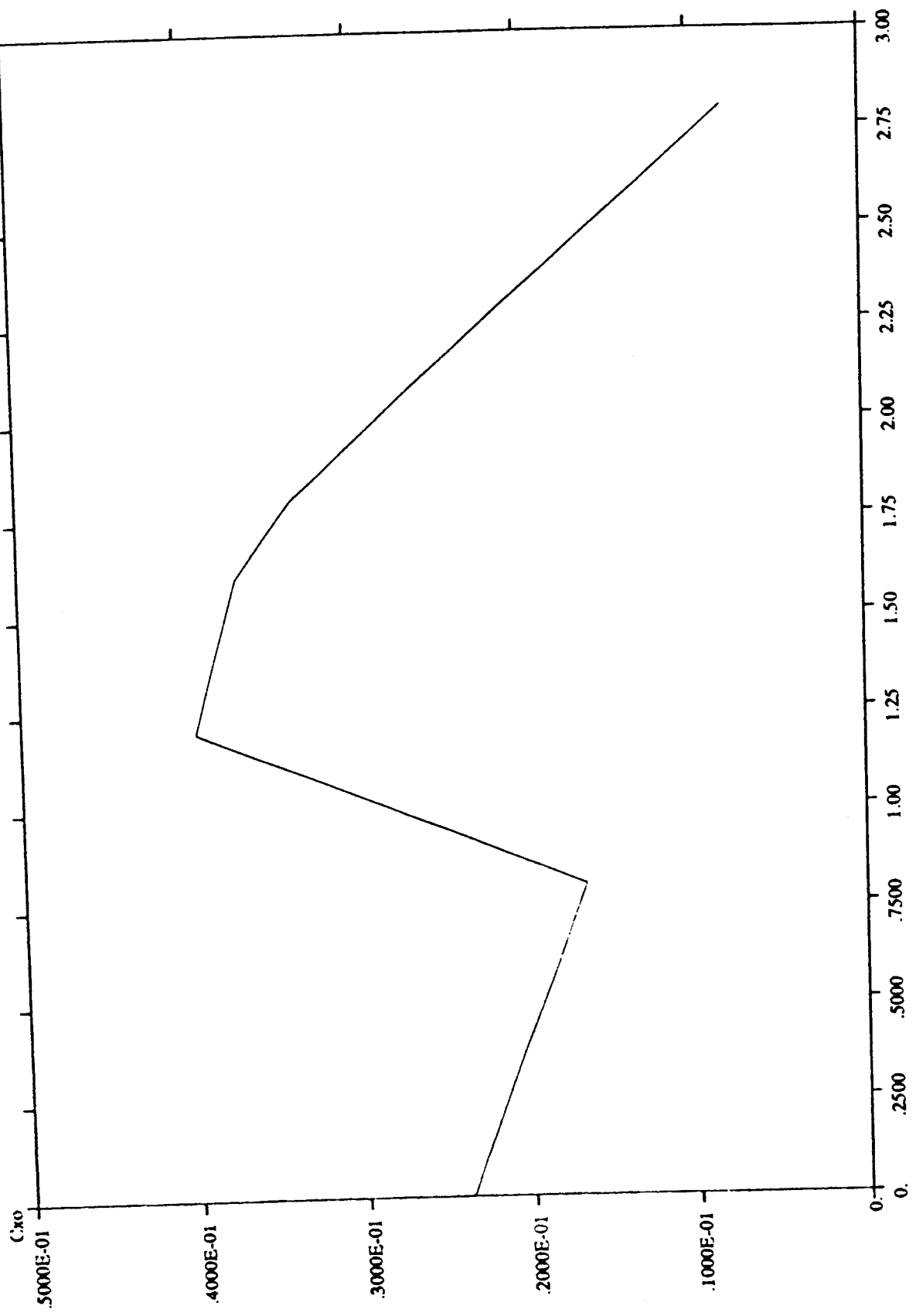


Exemple de Yes

FACTEURS DE CHARGE LIMITE

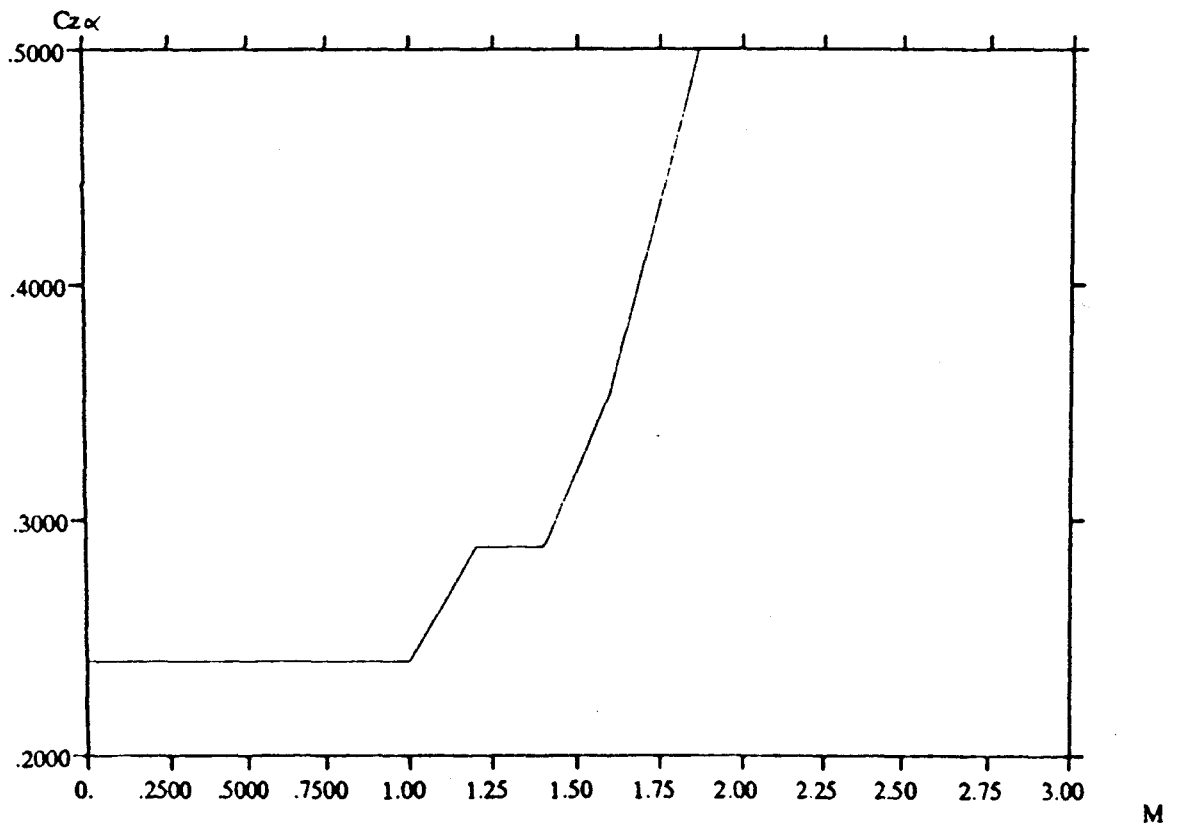
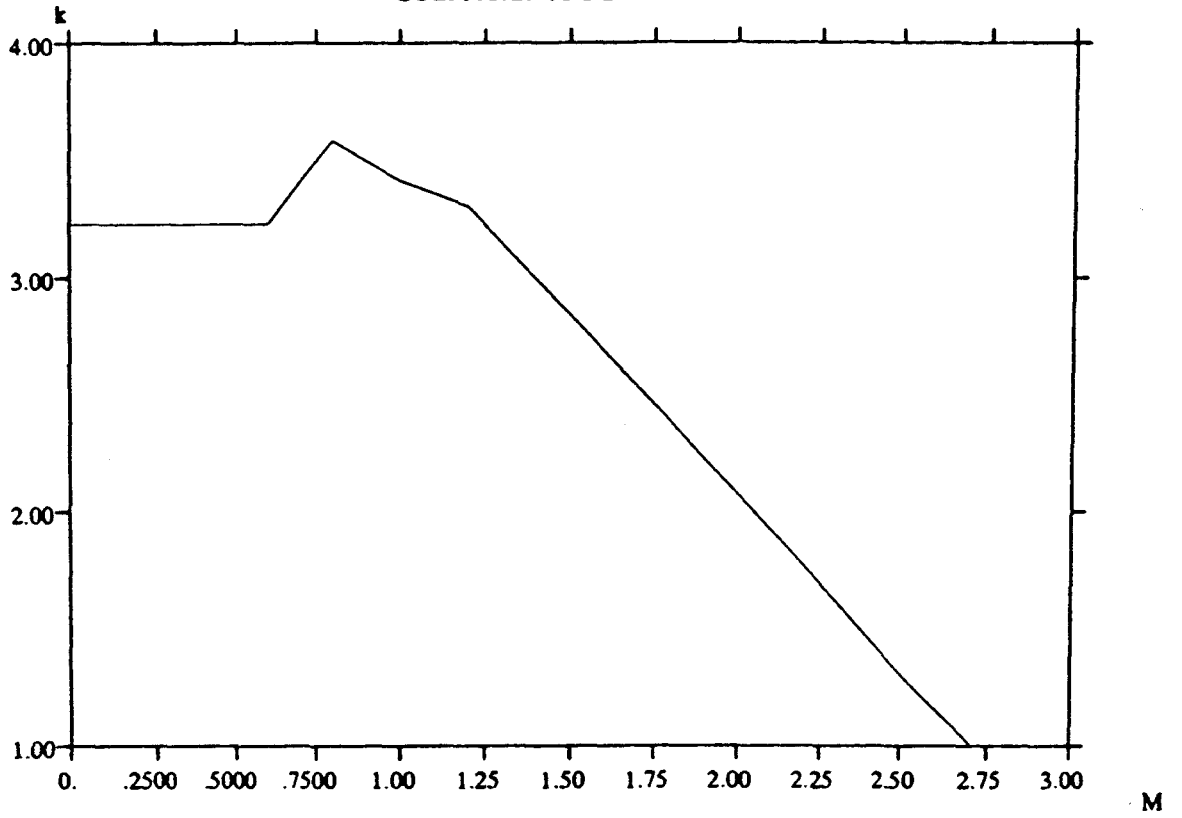


COEFFICIENT DE TRAINÉE

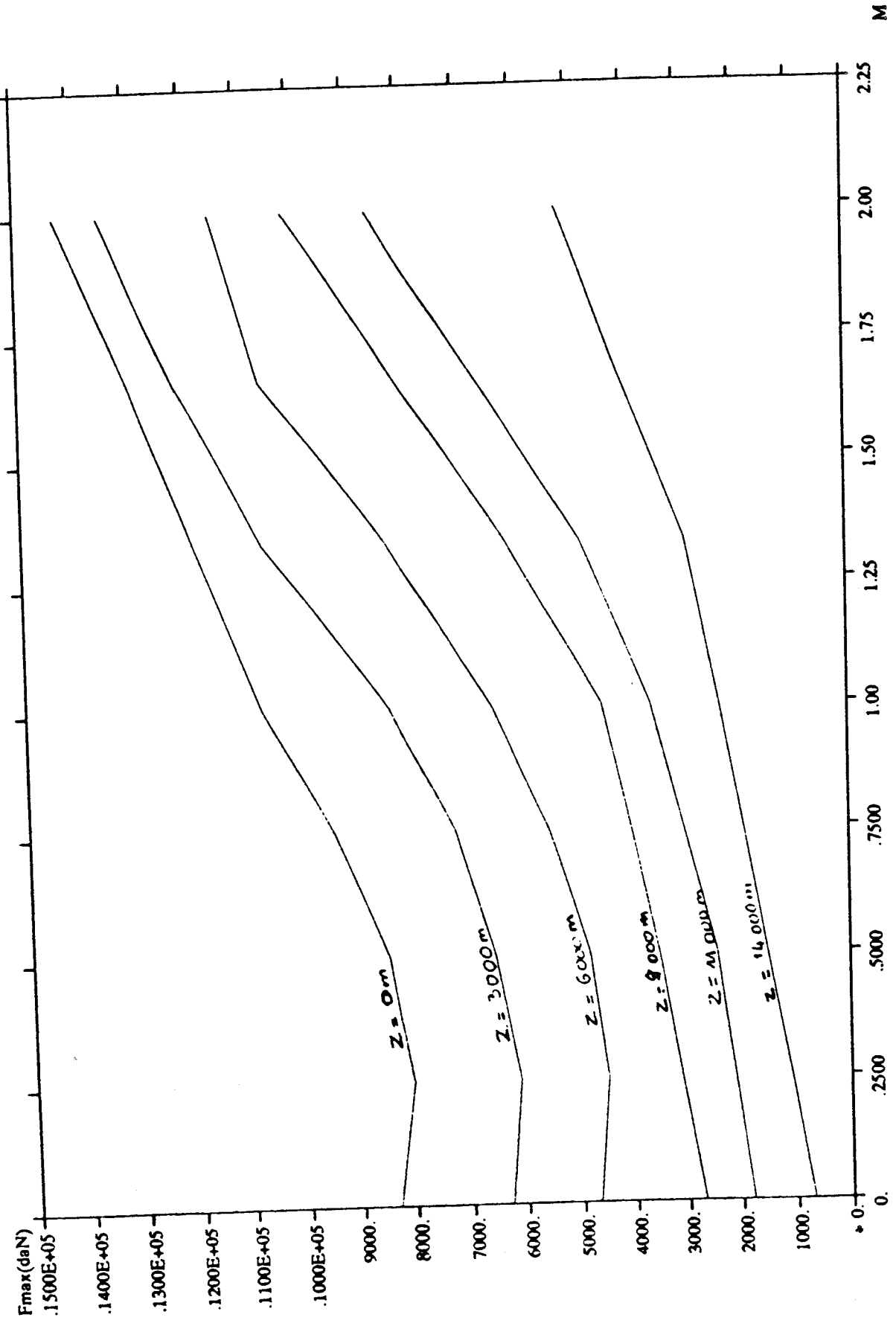


M

COEFFICIENTS DE PORTANCE

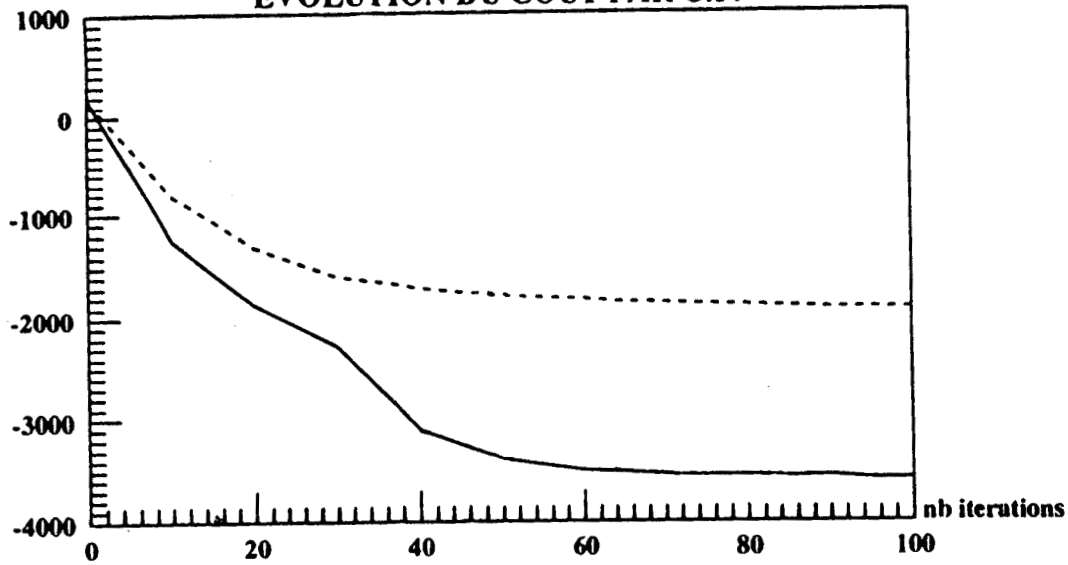


COURBES DE POUSSÉE PG-PC



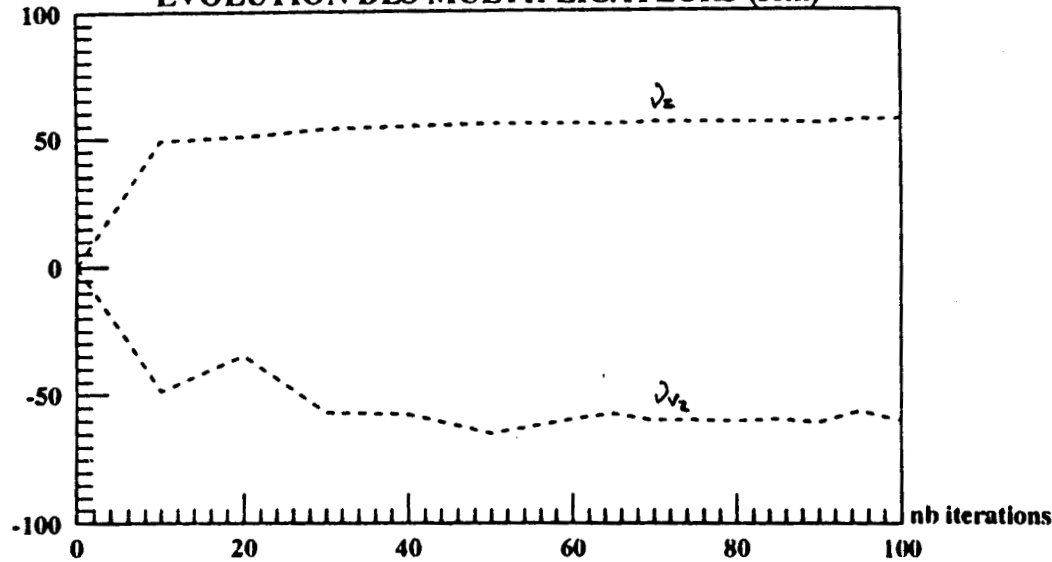


EVOLUTION DU COUT PAR G.P.

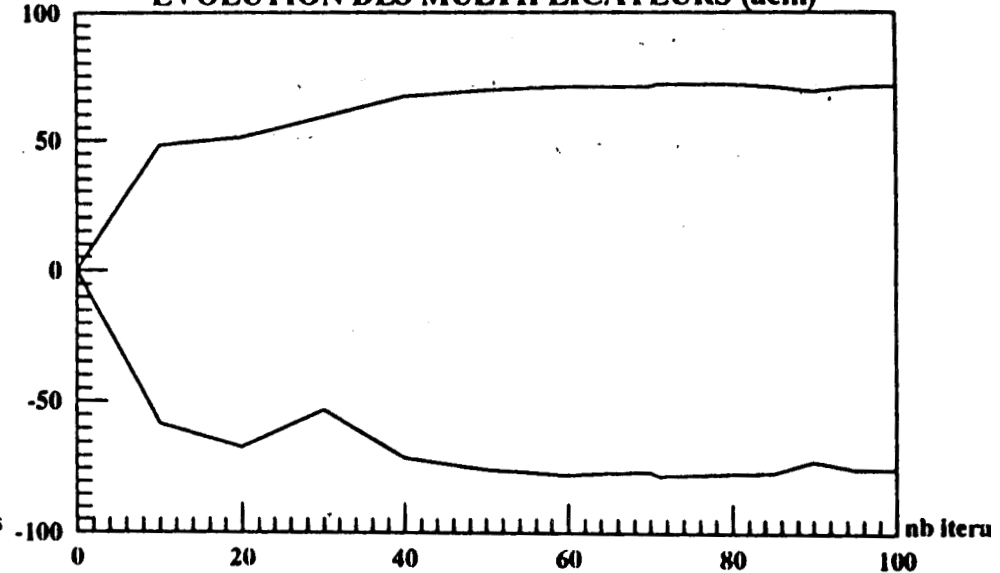


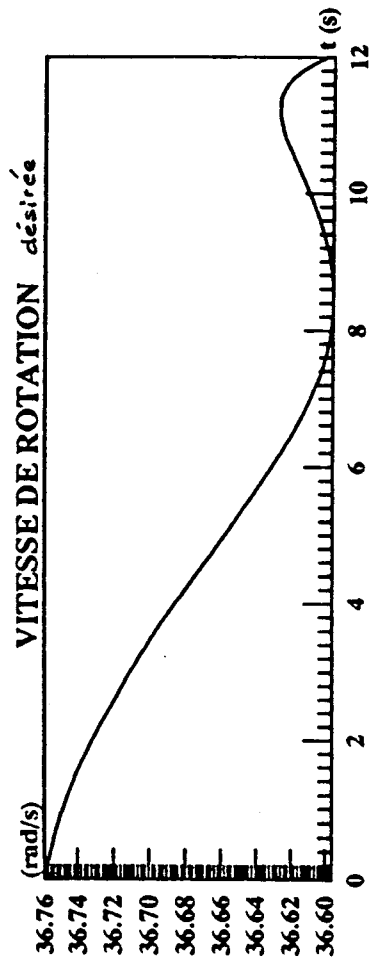
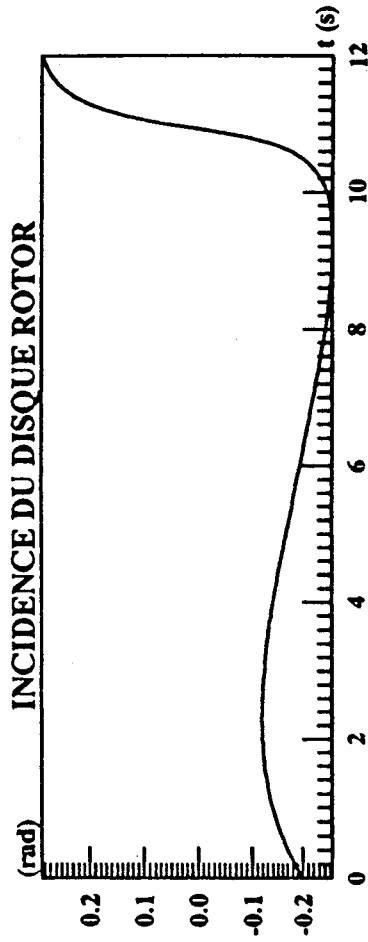
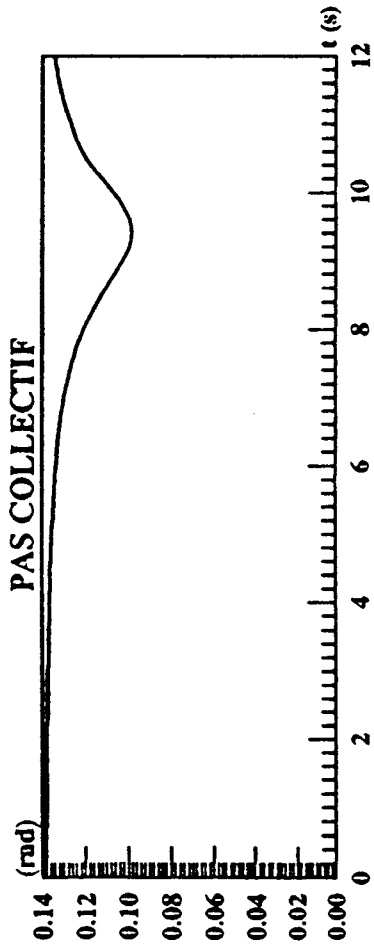
.... sans contrôle moteur
 — avec contrôle moteur

EVOLUTION DES MULTIPLICATEURS (scm)



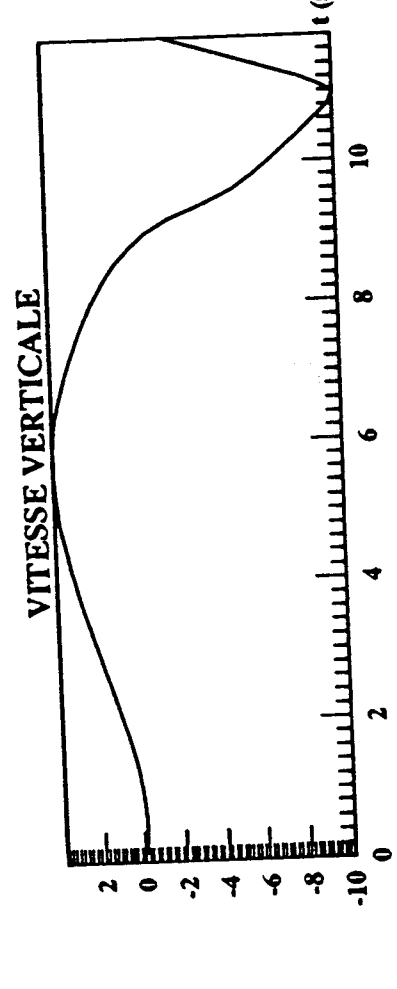
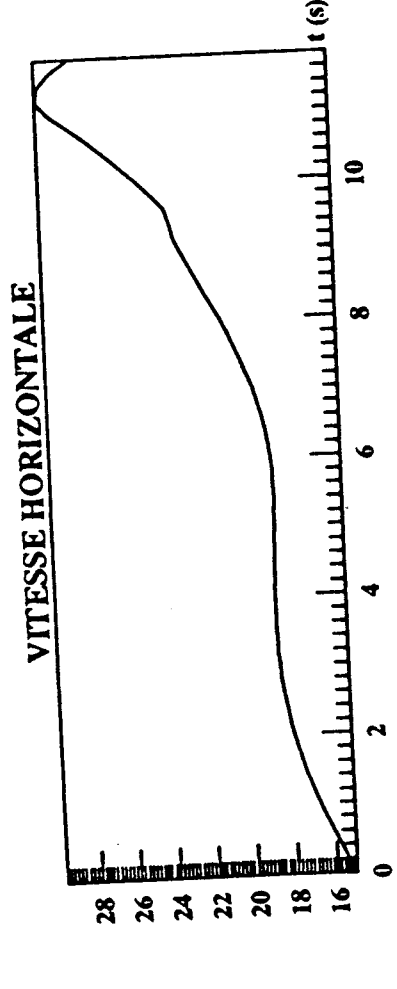
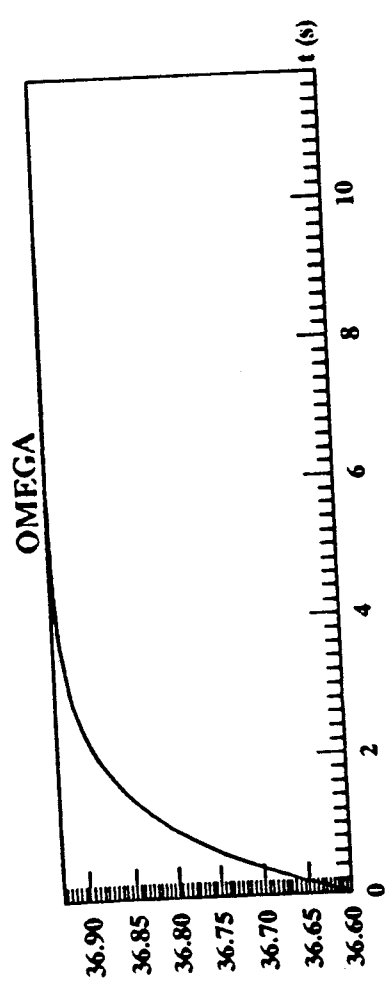
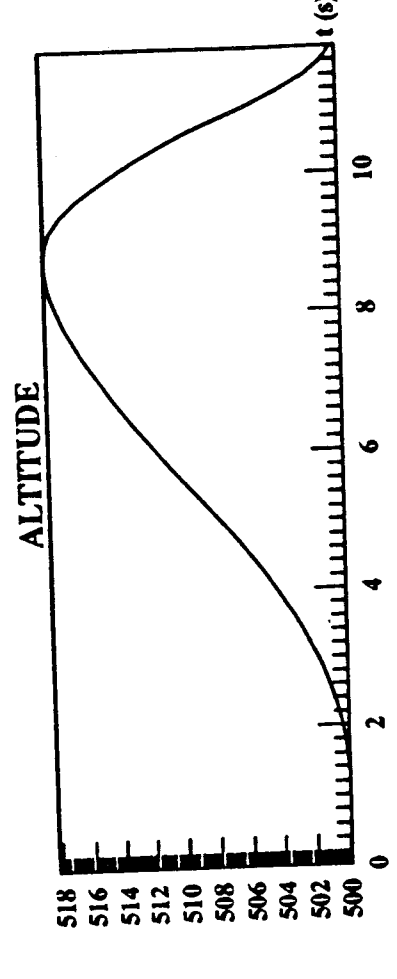
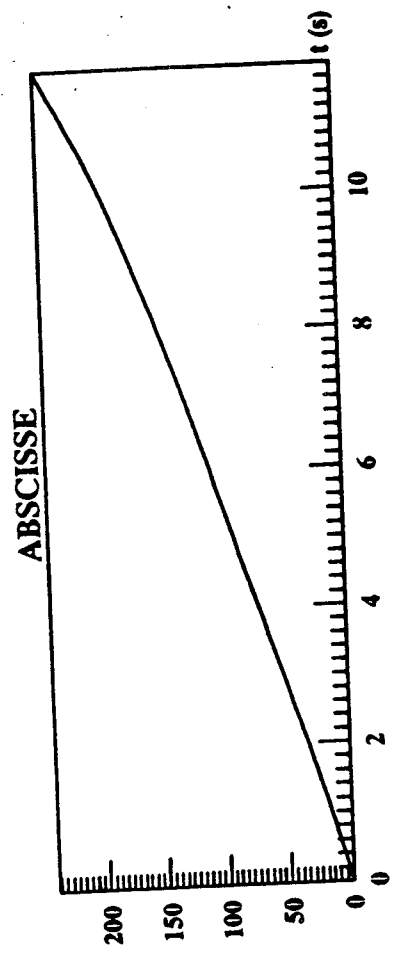
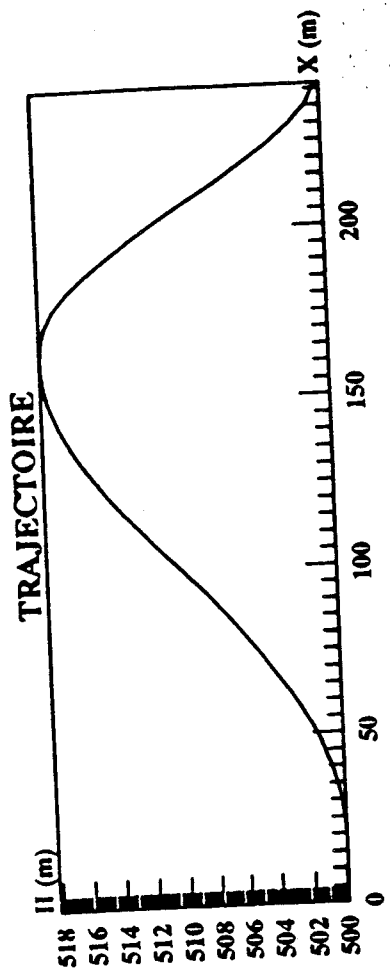
EVOLUTION DES MULTIPLICATEURS (acm)



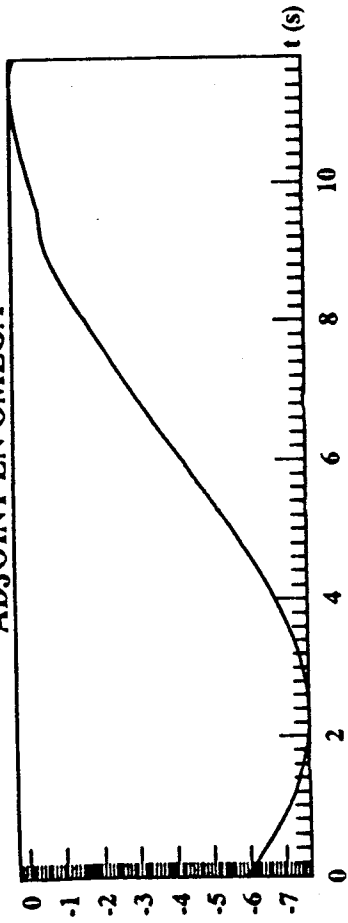


CAS sans contrôle moteur
(vitesse de rotation limitée à 11%
de la vitesse nominale)

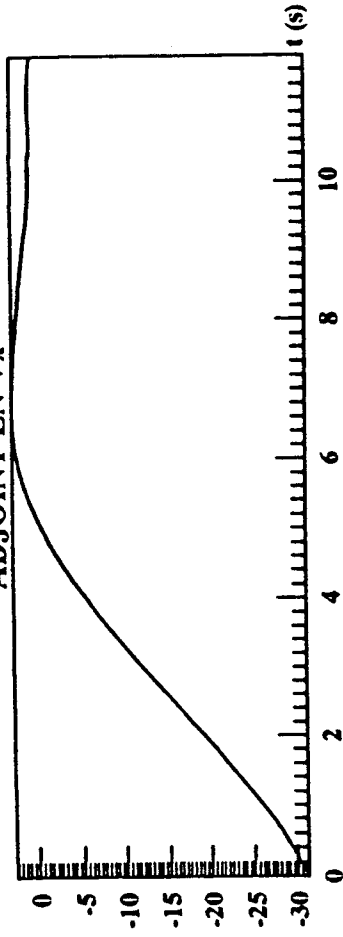
Résultat de 30 itérations par GP



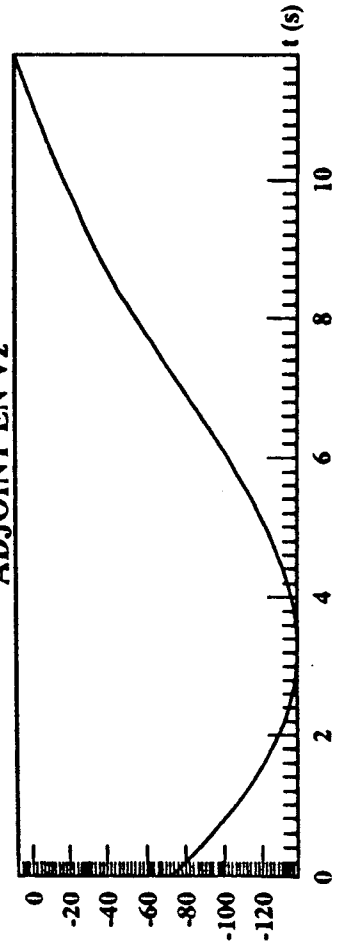
ADJOINT EN OMEGA



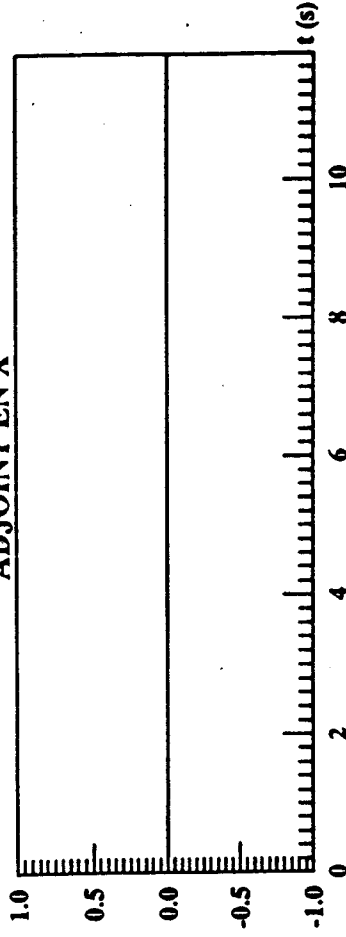
ADJOINT EN Vx



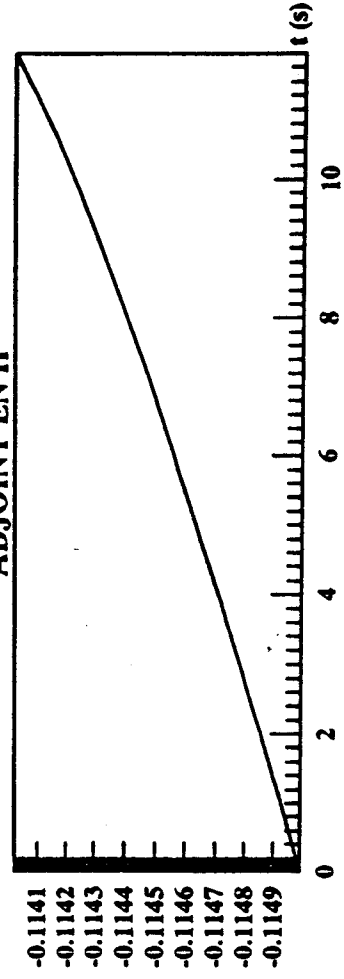
ADJOINT EN Vz

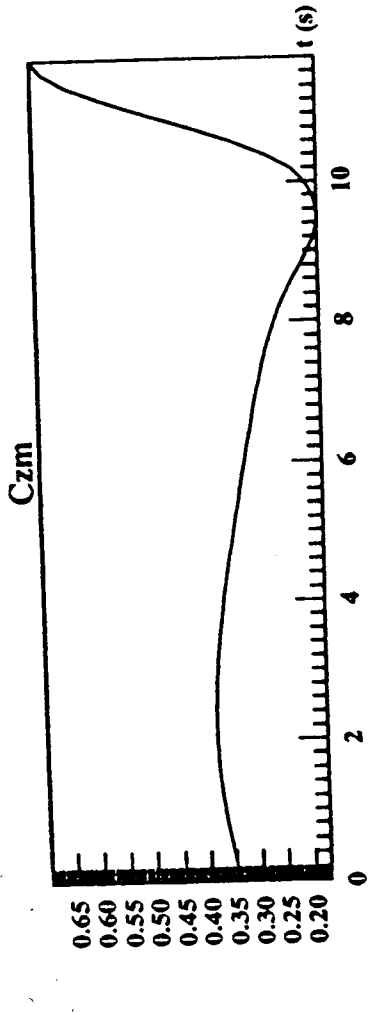
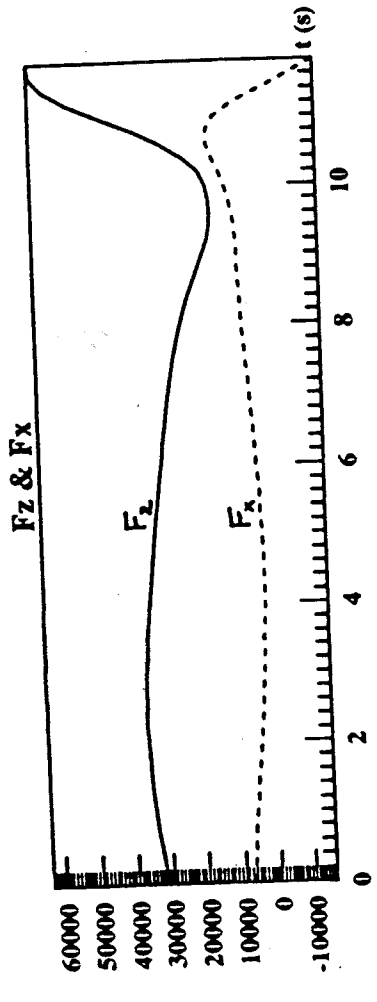
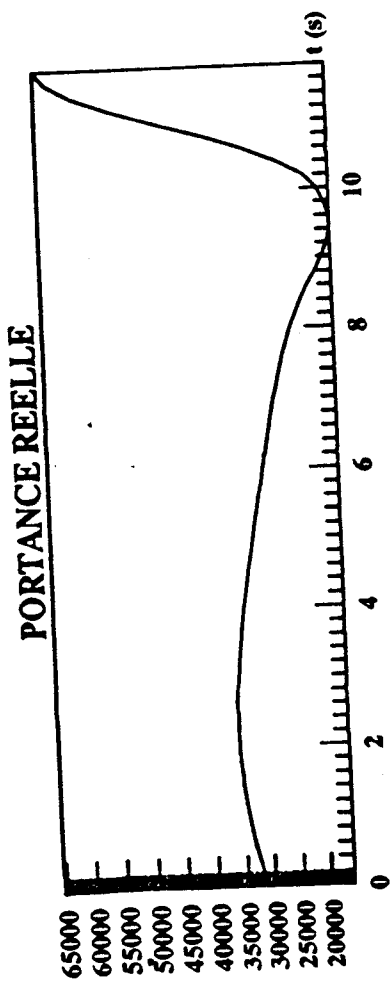
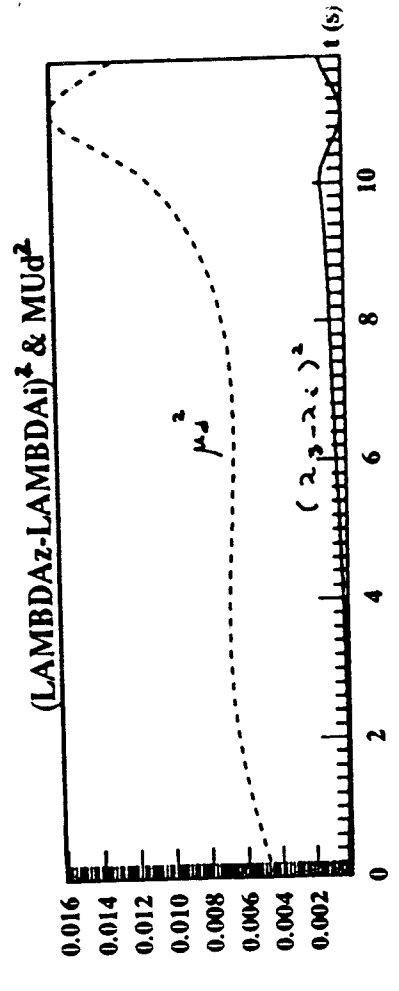
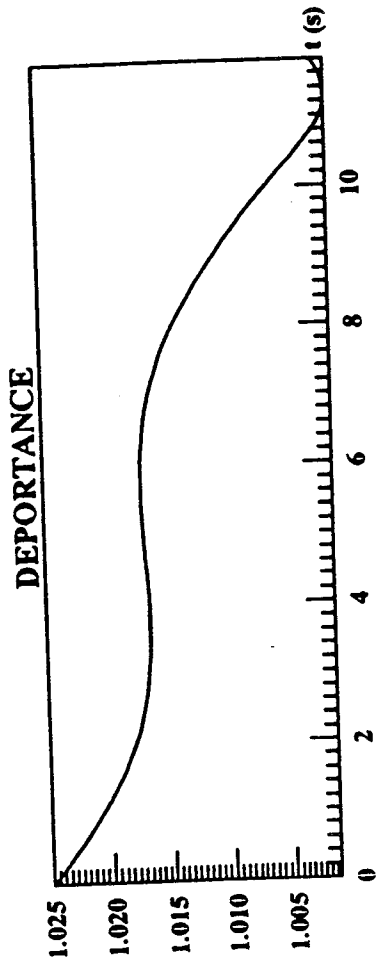
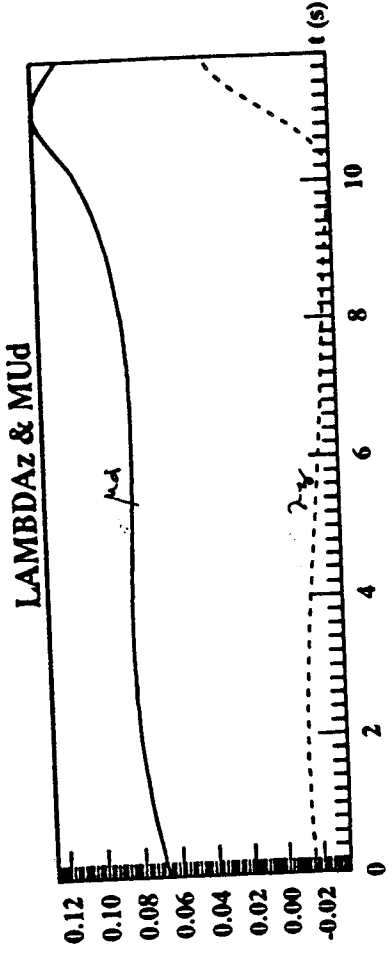


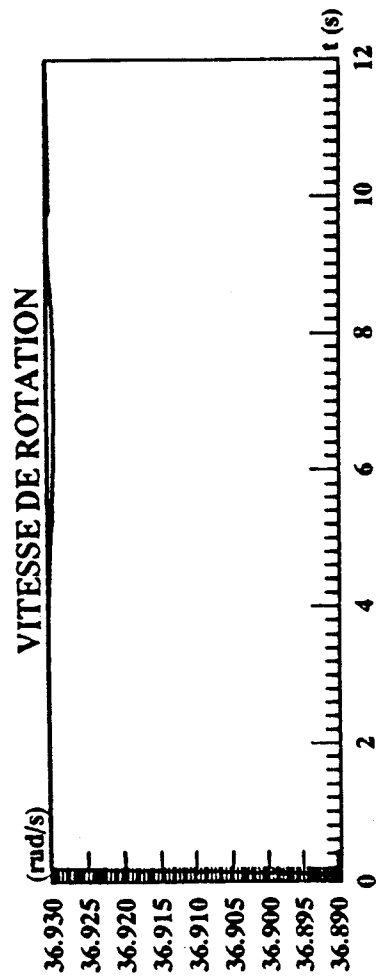
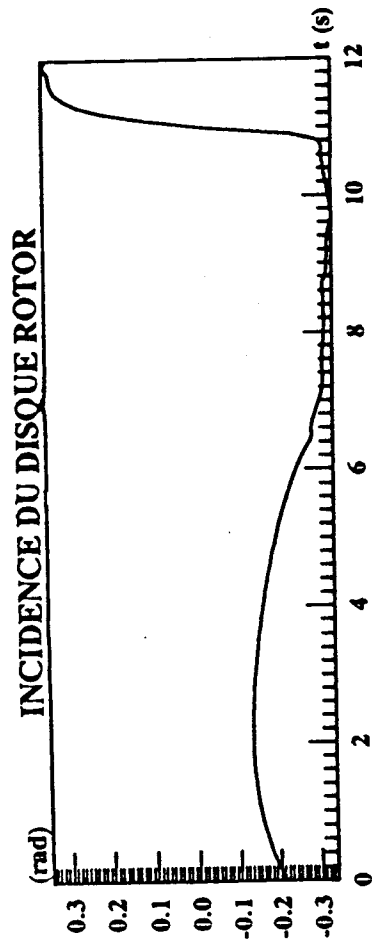
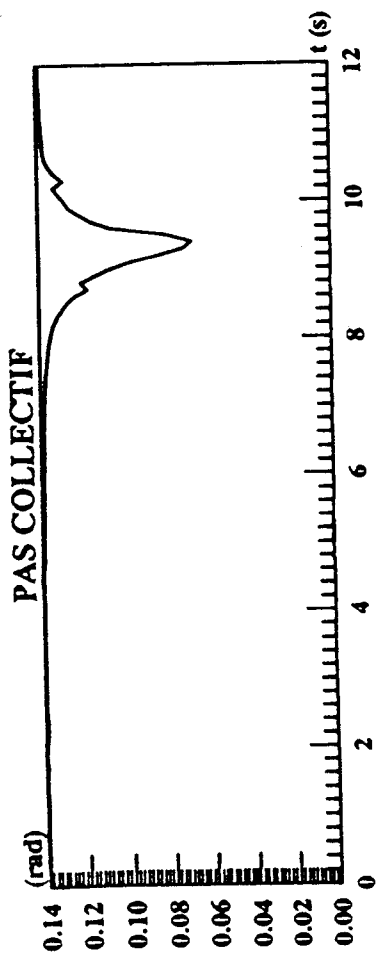
ADJOINT EN X



ADJOINT EN H

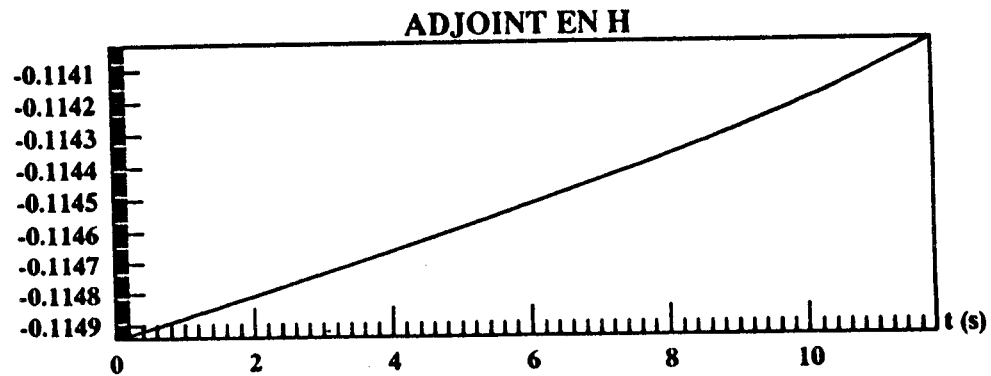
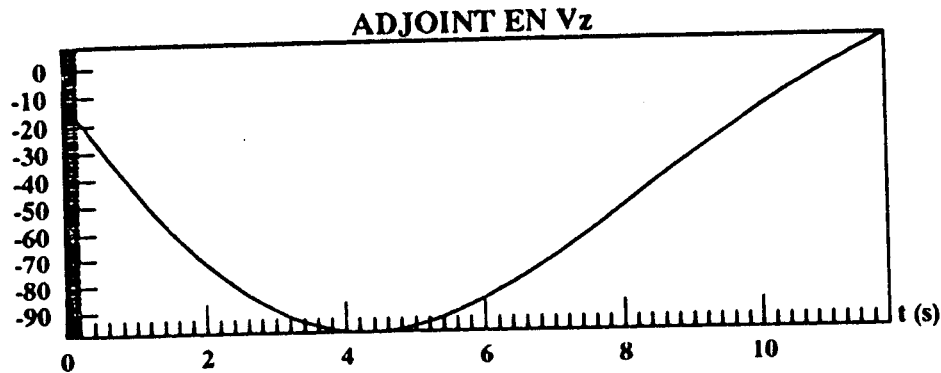
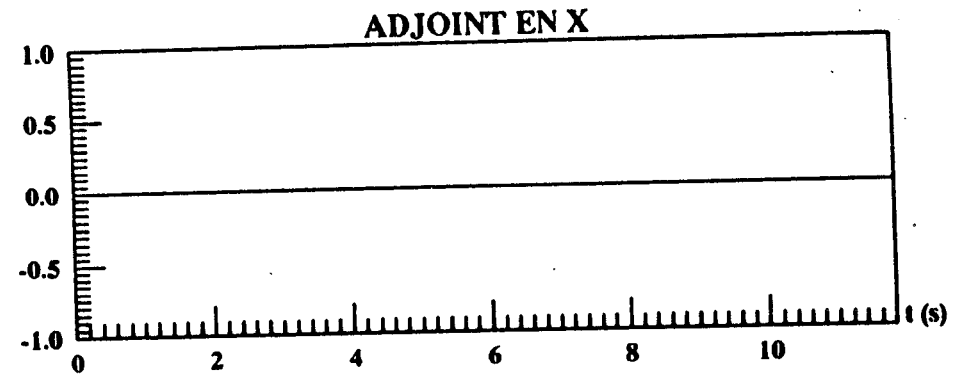
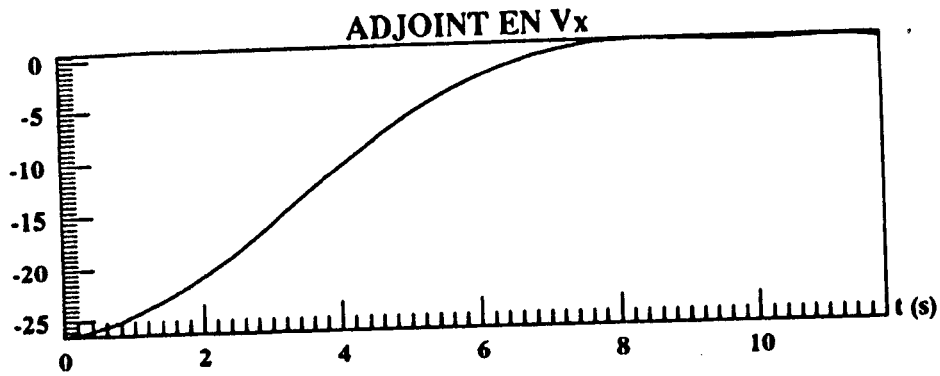
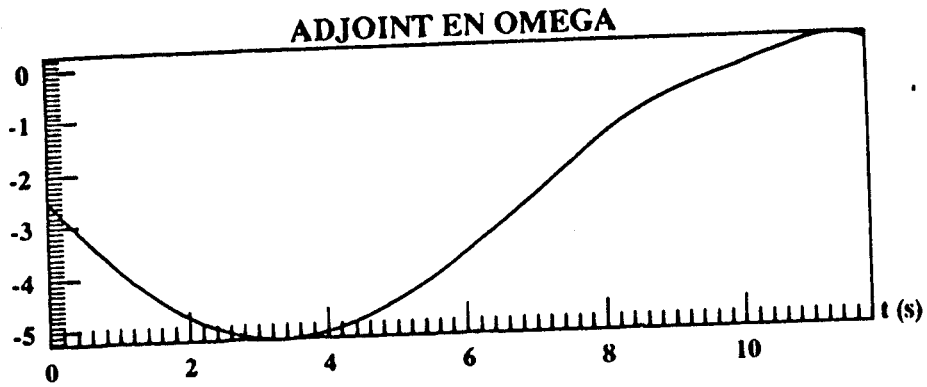


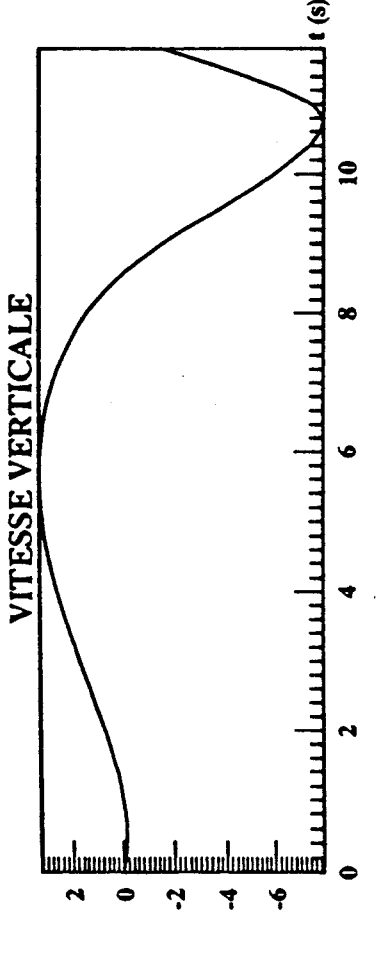
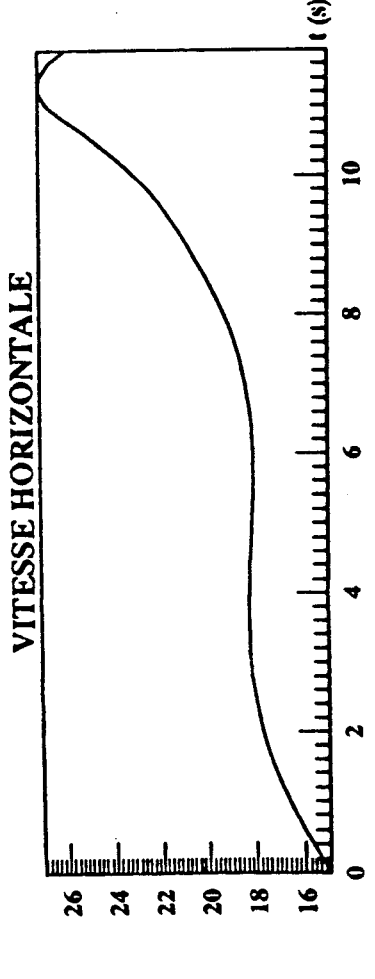
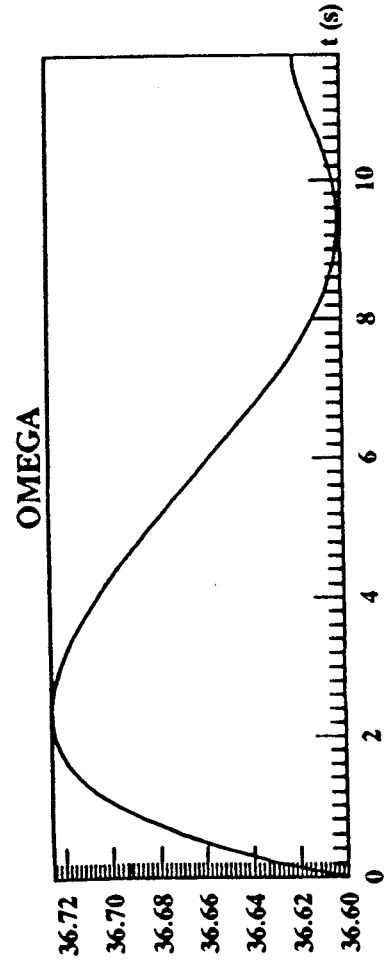
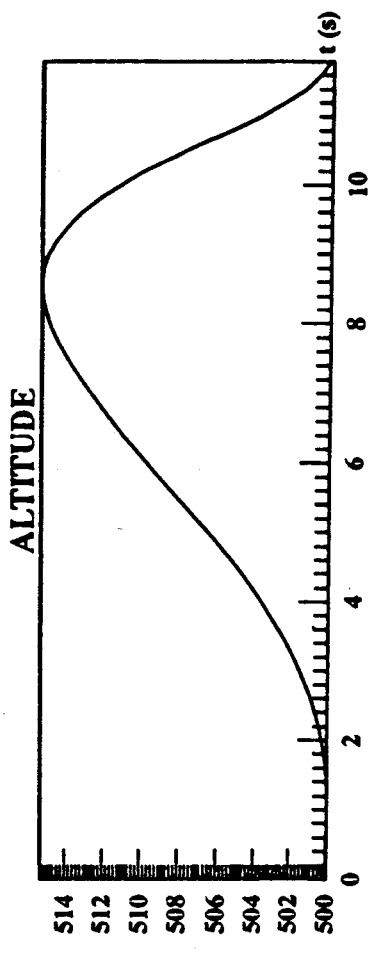
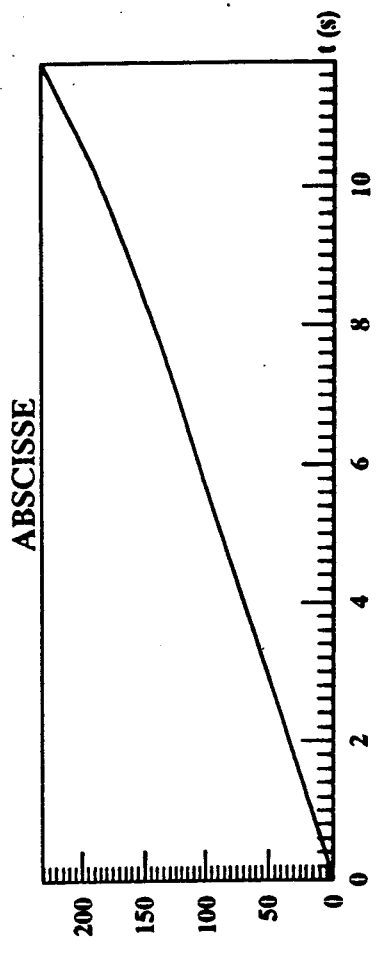
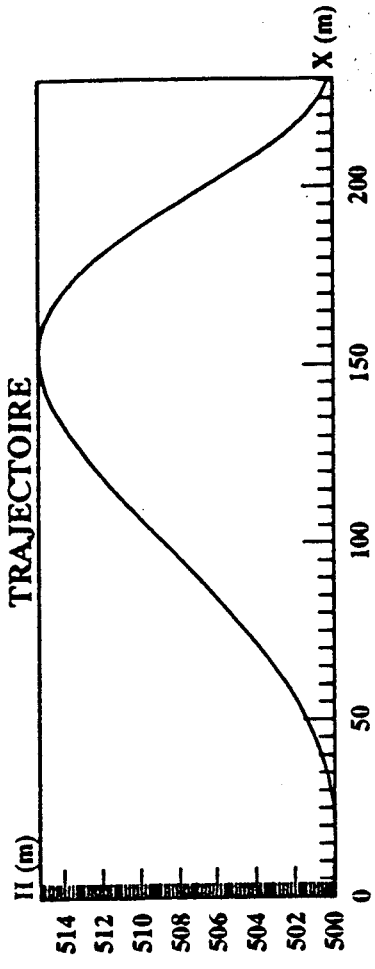




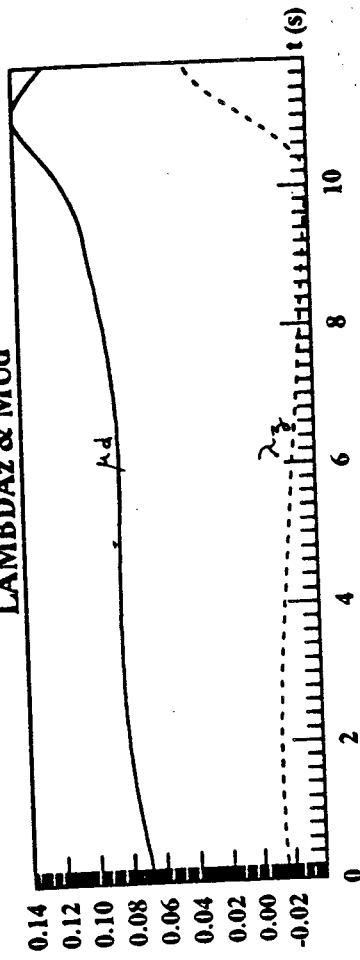
Cas sans controle moteur

Résultat de 2 itérations par QSL
 Initialisation avec les courbes précédentes

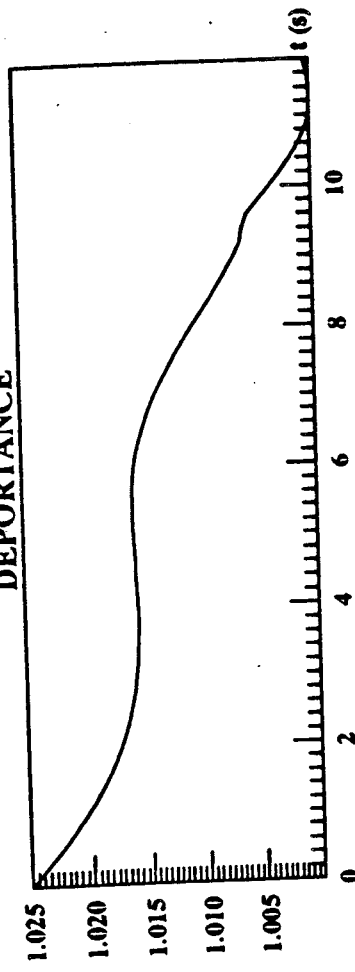




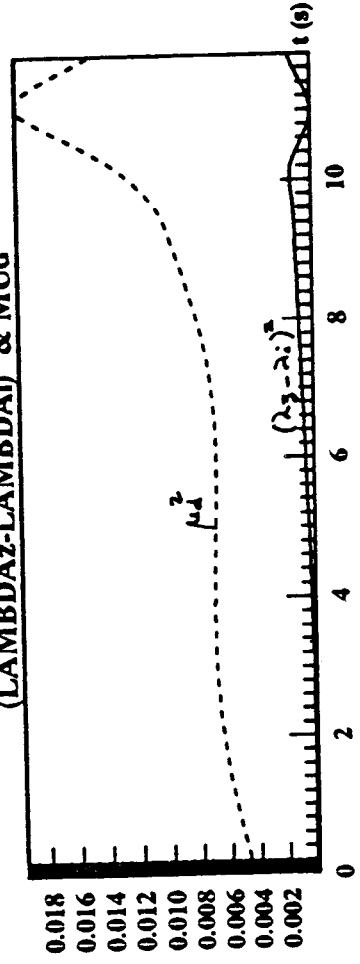
LAMBDAz & MUd



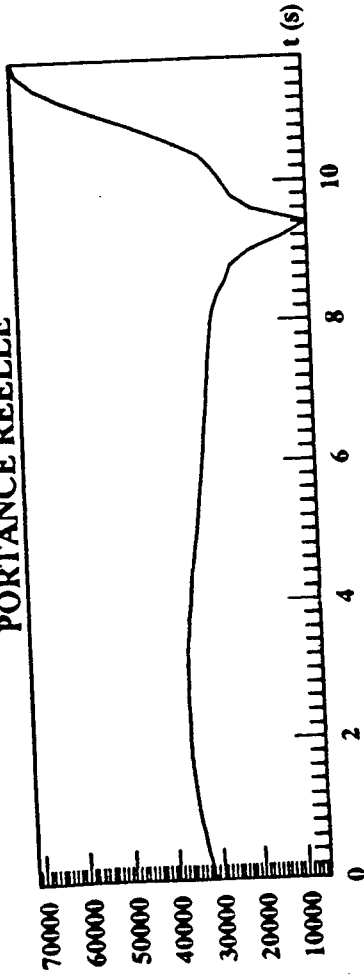
DEPORTANCE



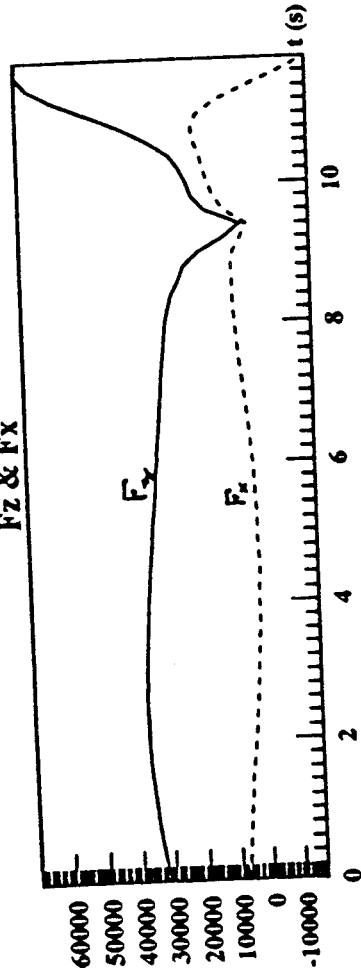
(LAMBDAz-LAMBDAj)² & MUd²



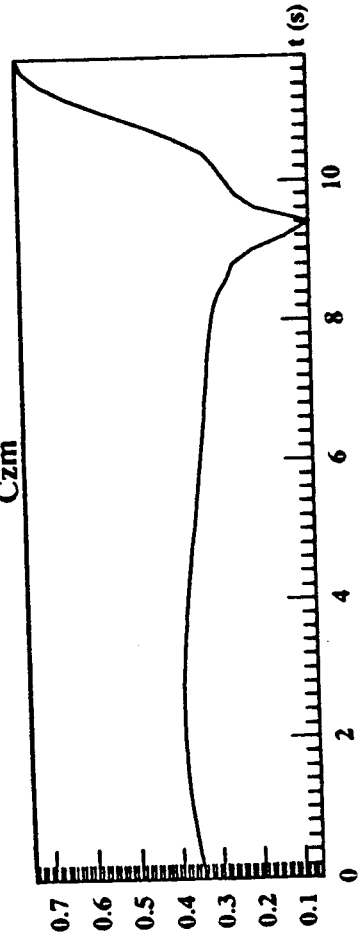
PORTANCE REELLE

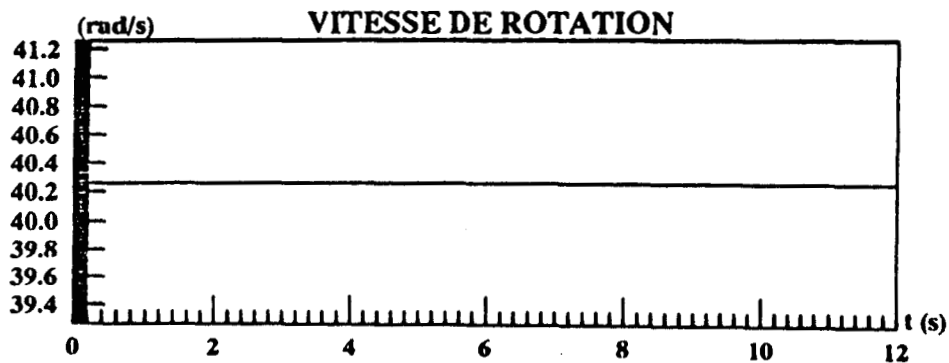
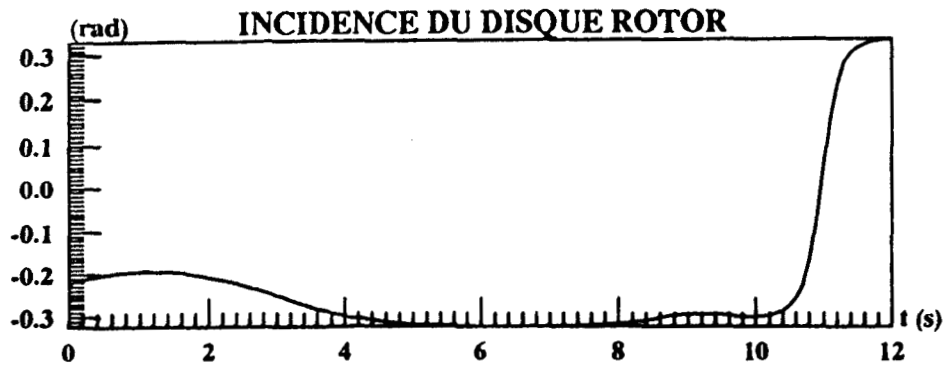
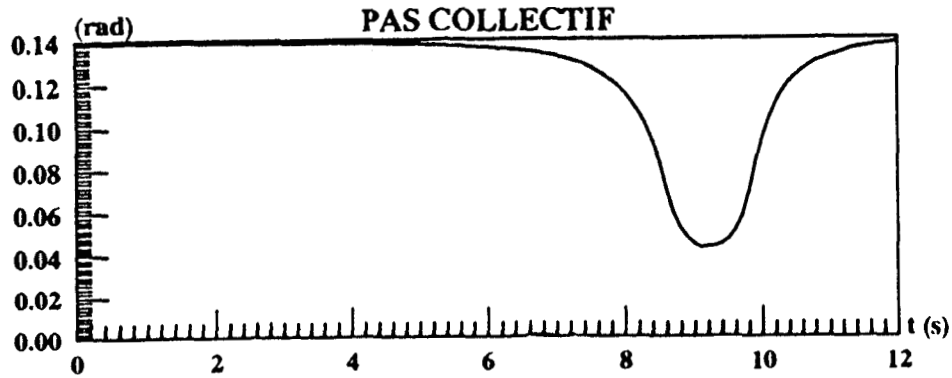


Fz & Fx



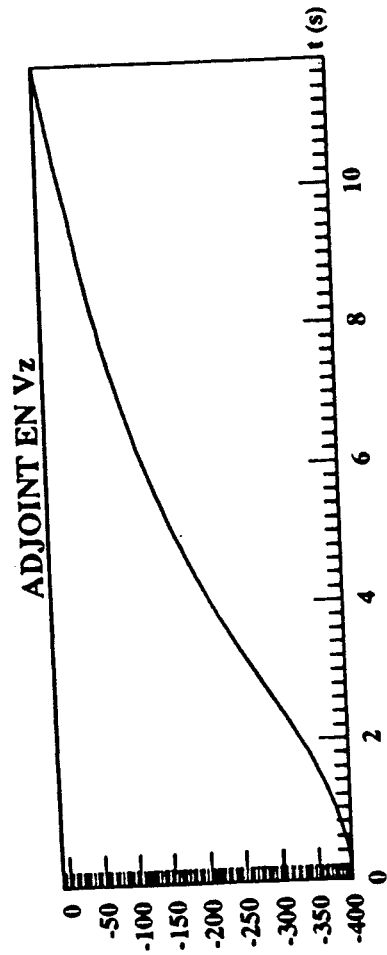
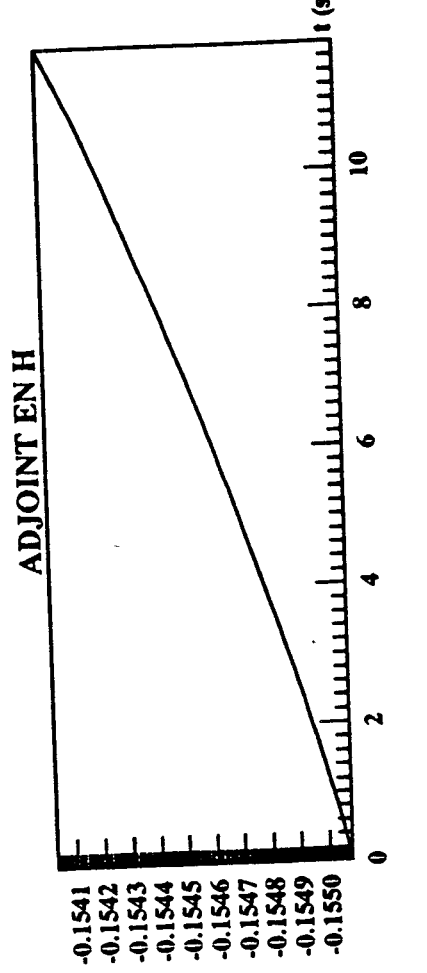
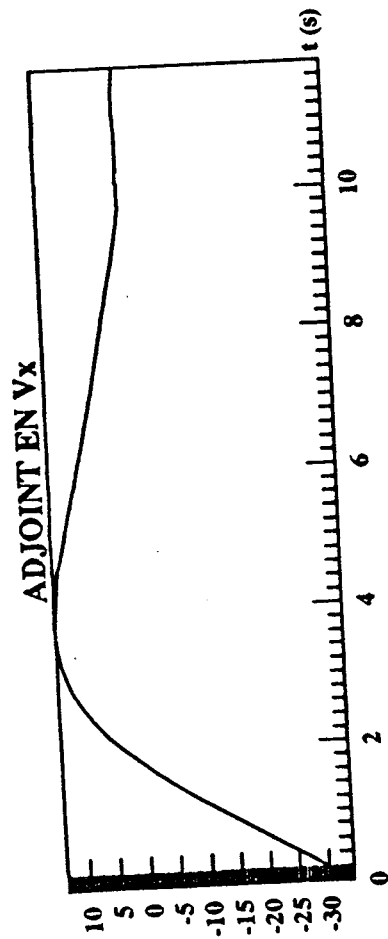
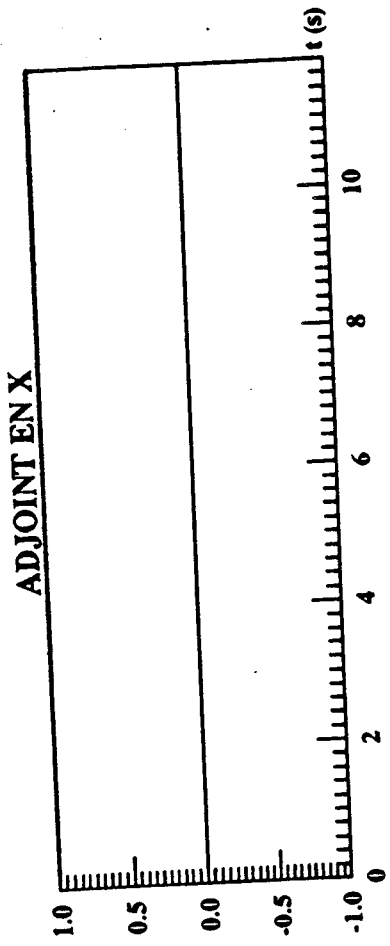
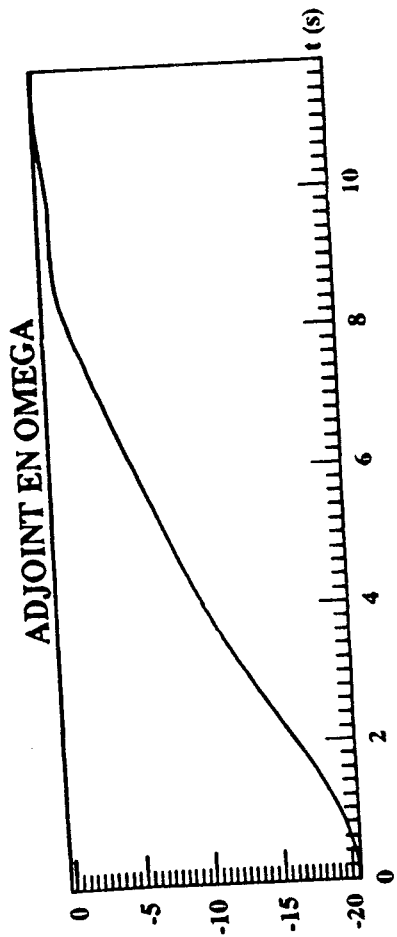
Czm

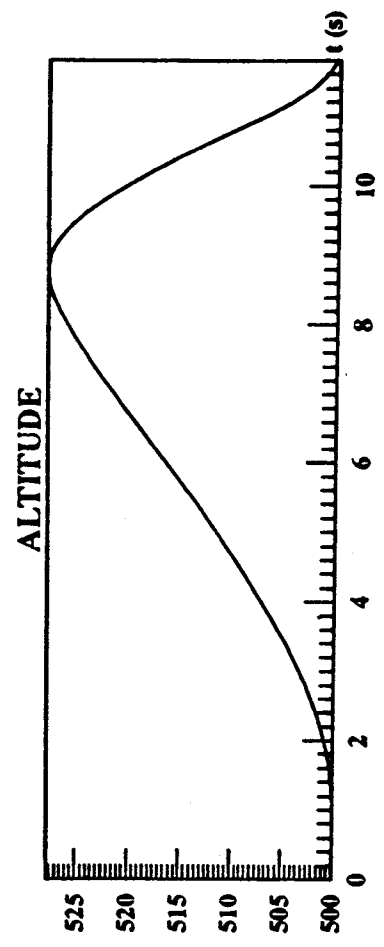
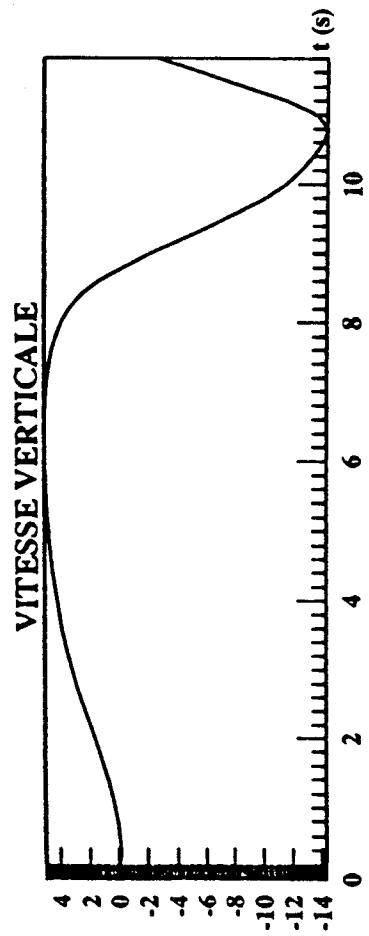
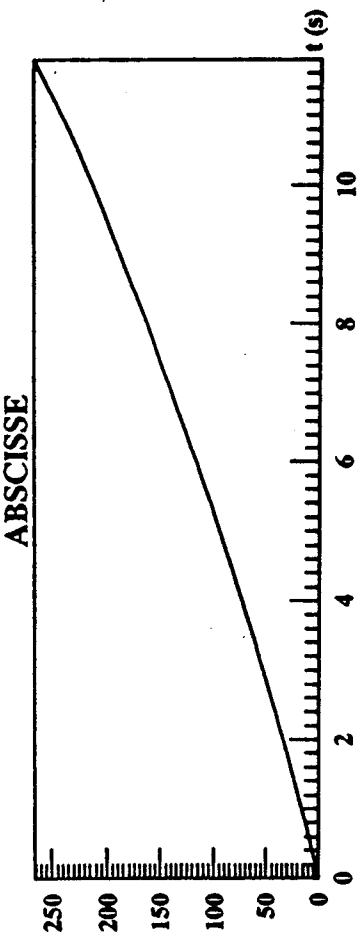
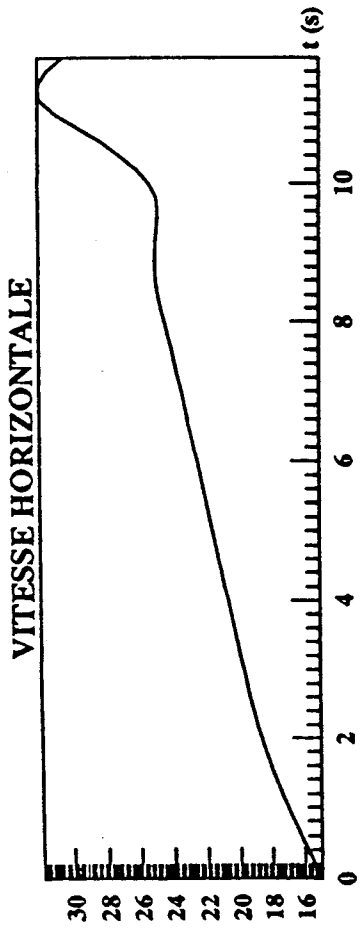
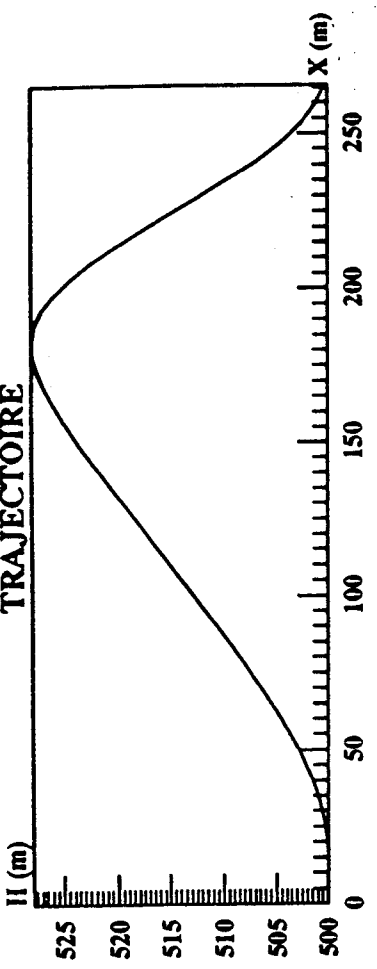




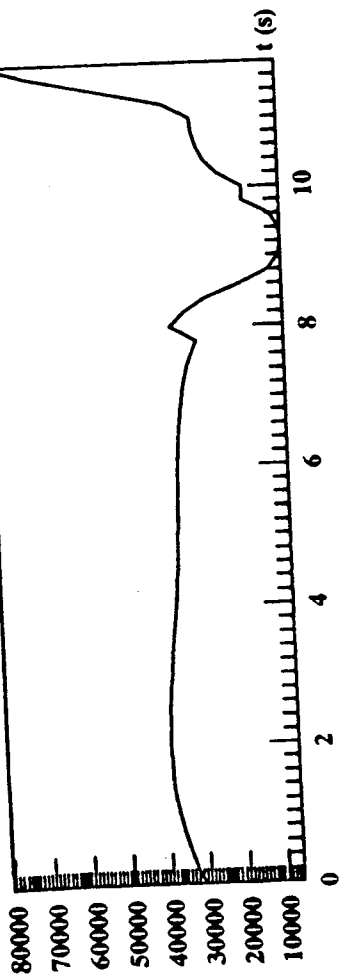
cas avec contrôle moteur

Résultat de 50 itérations par GP

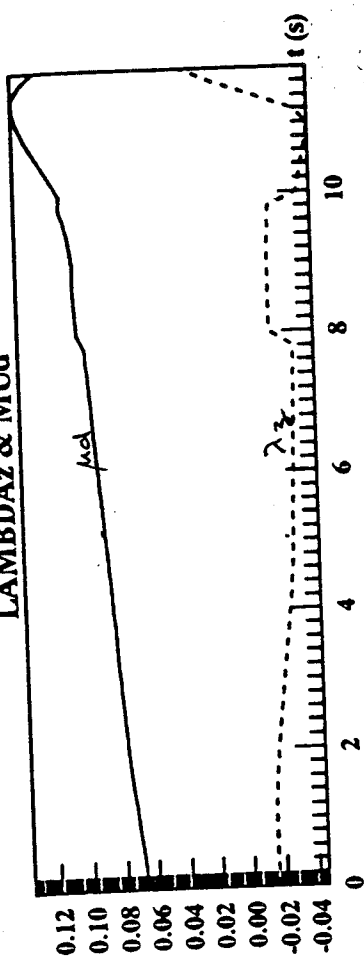




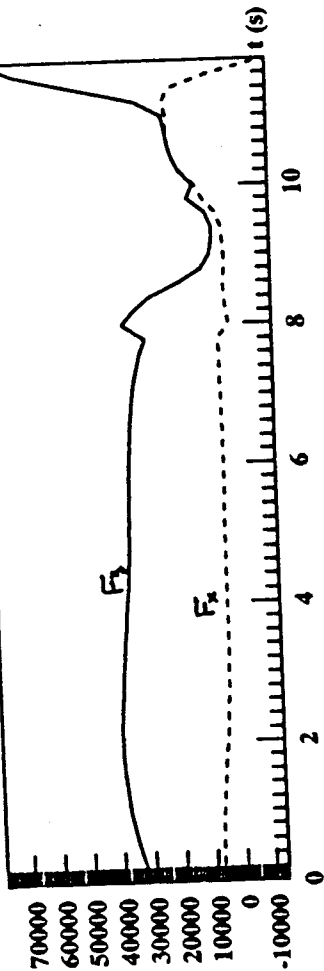
PORTANCE REELLE



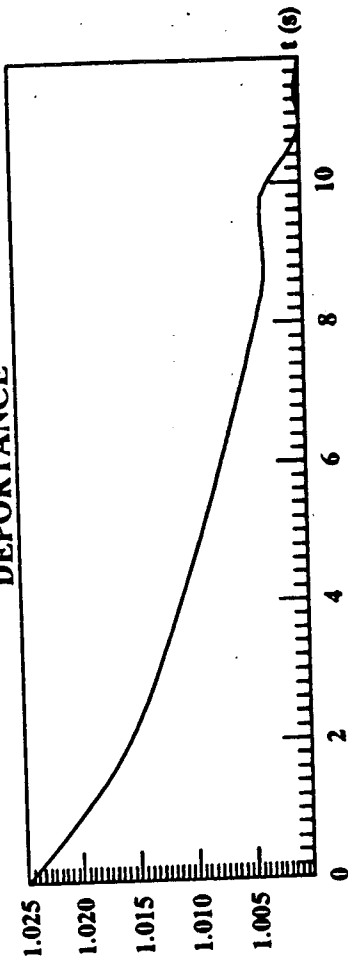
LAMBDAz & MUd



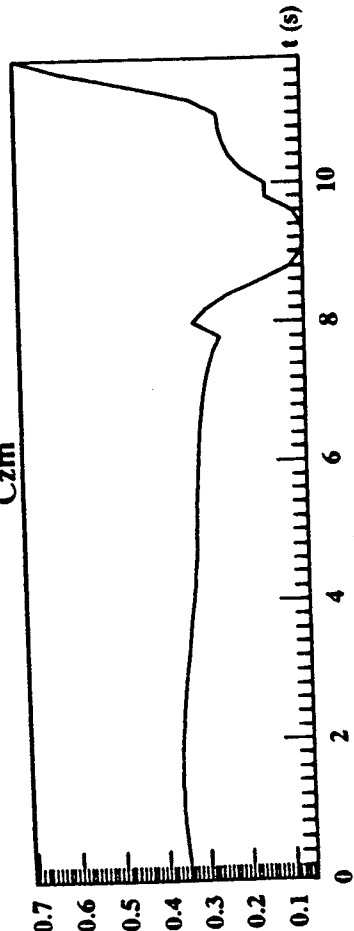
Fz & Fx



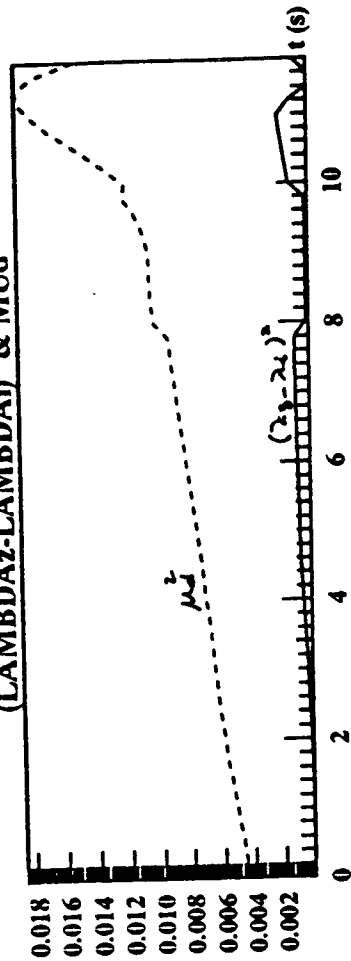
DEPORTANCE

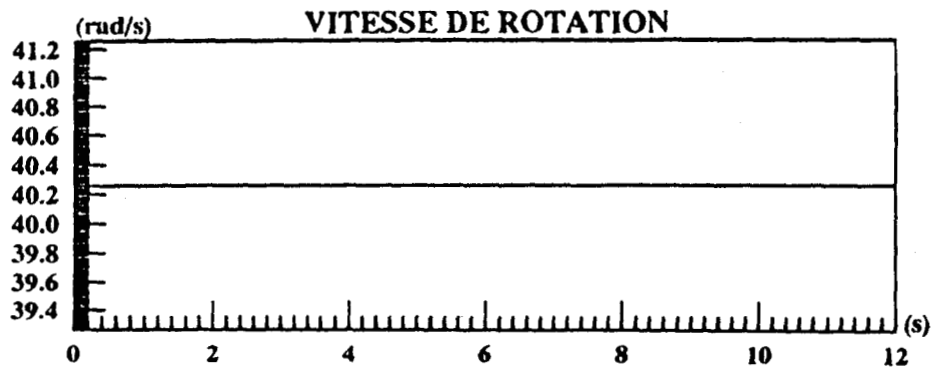
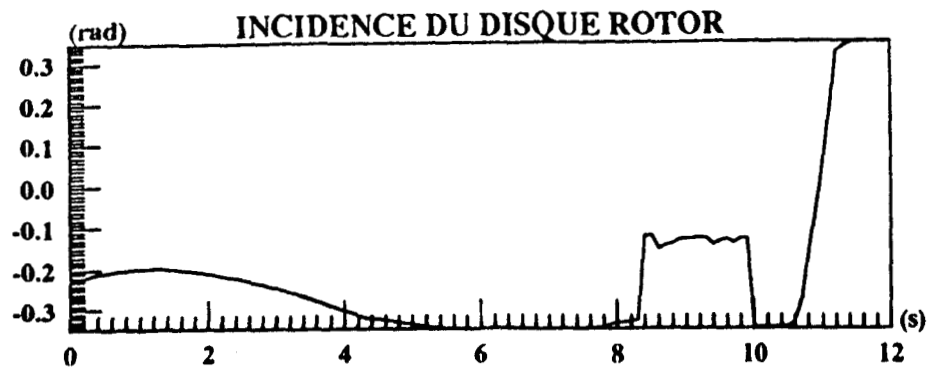
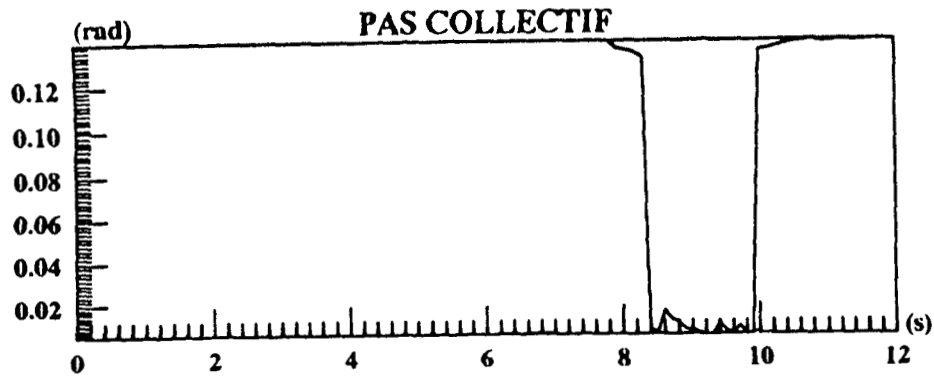


Czm



(LAMBDAz-LAMBDAi)^2 & MUd^2

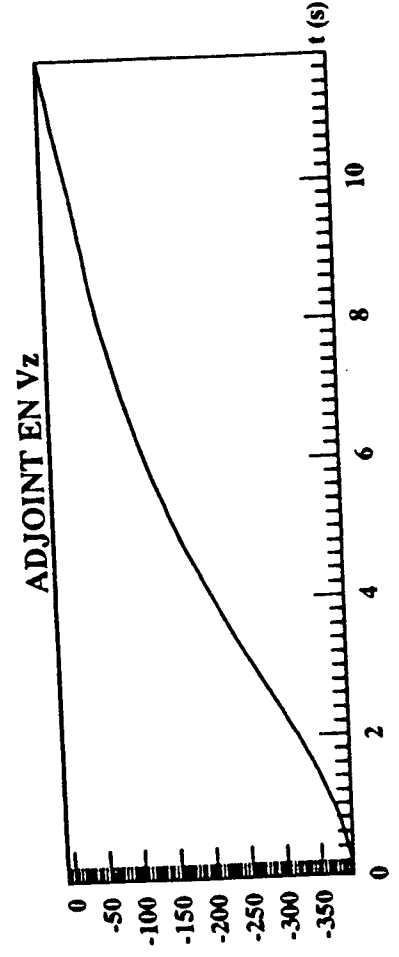
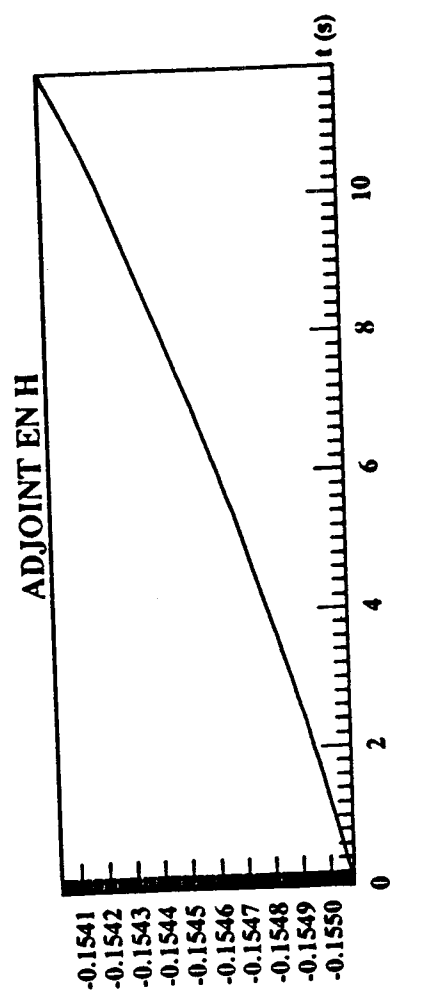
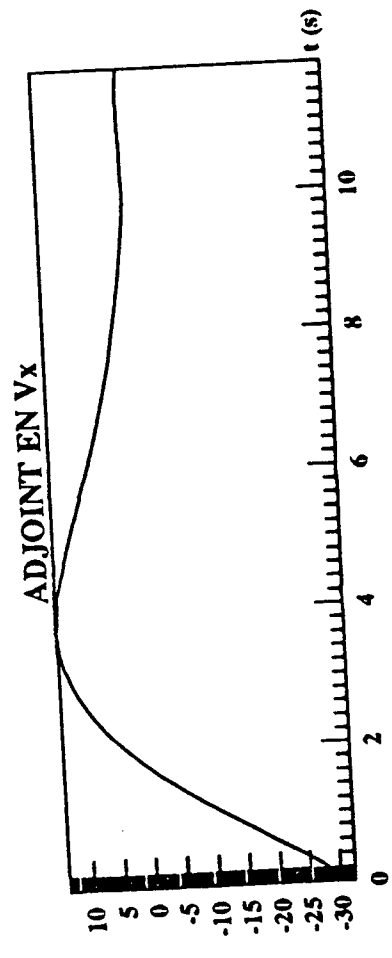
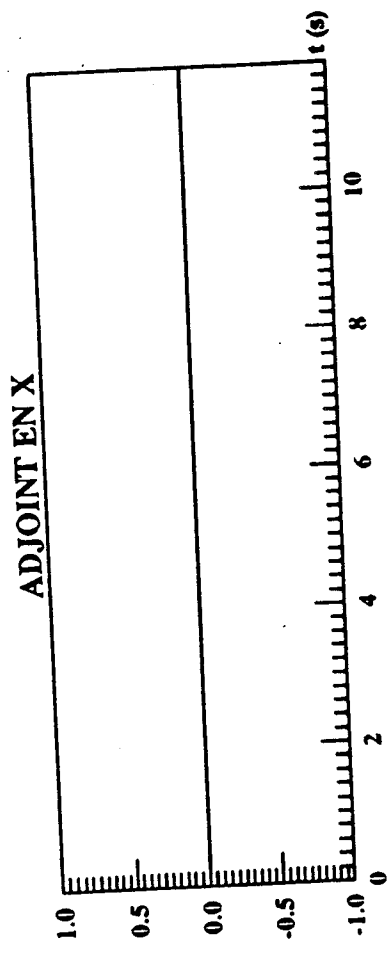
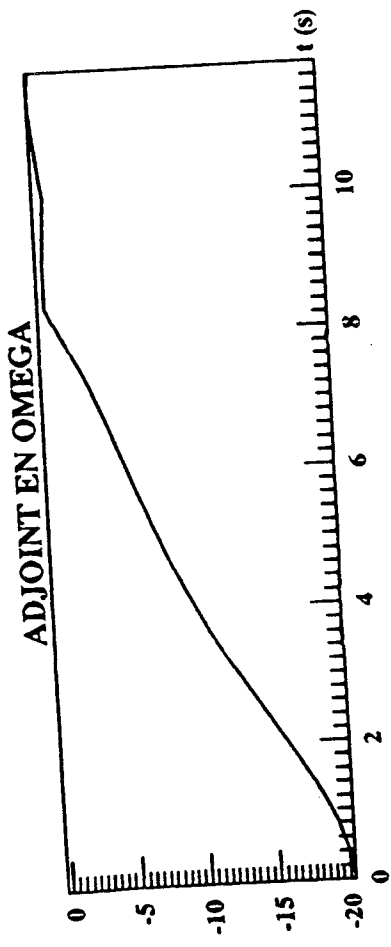


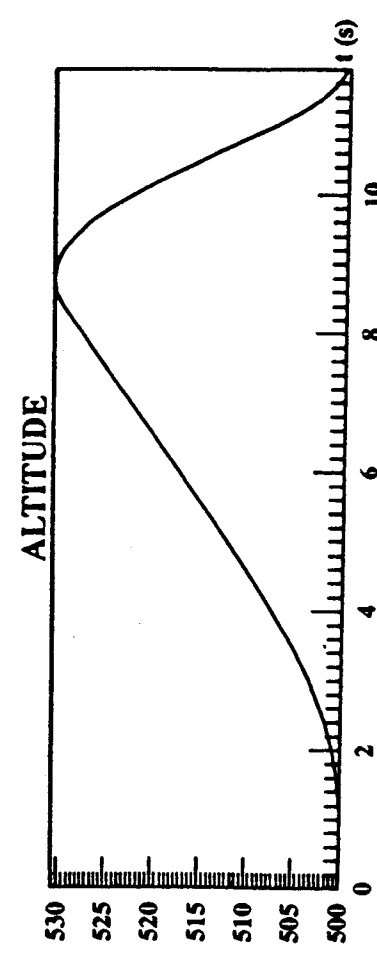
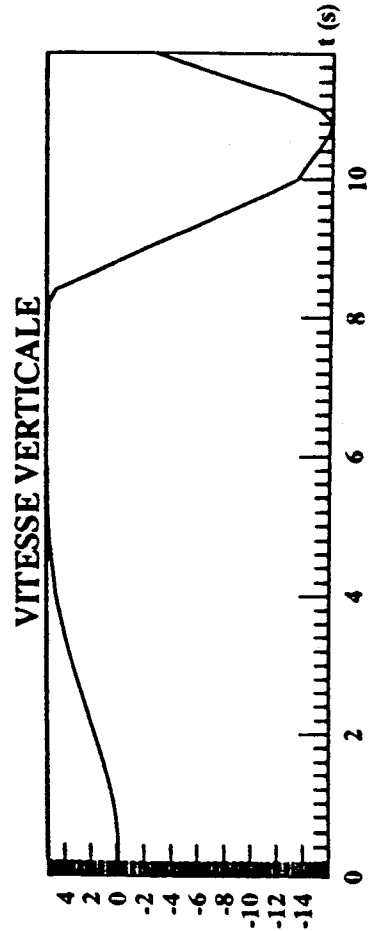
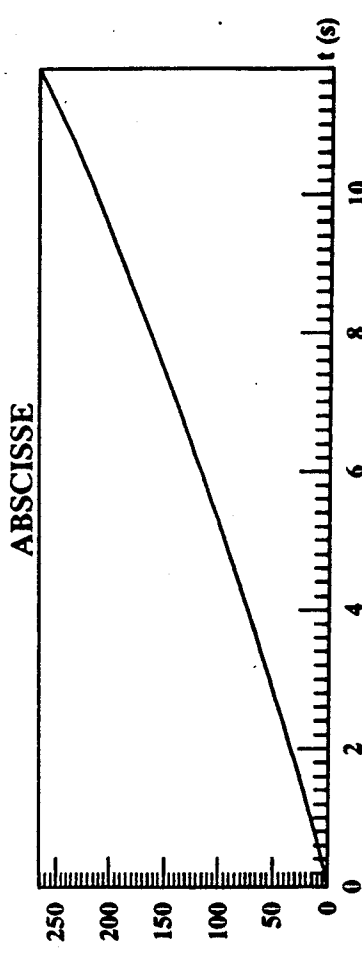
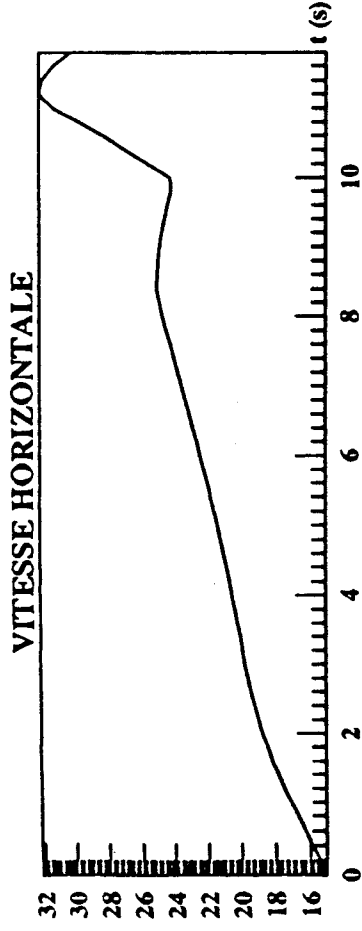
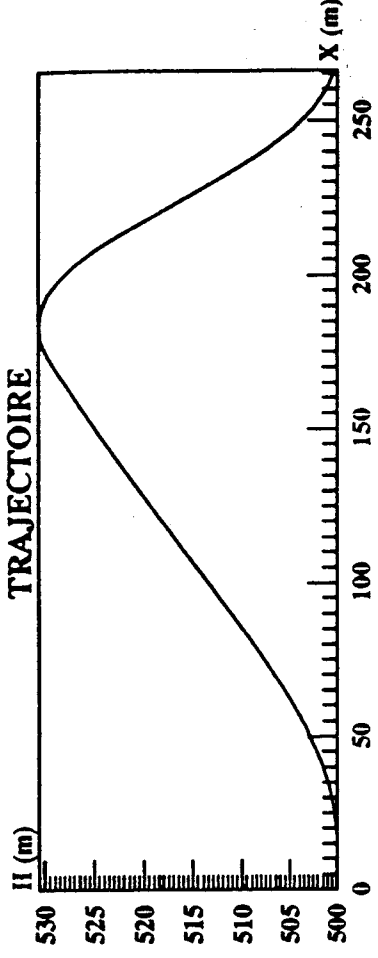
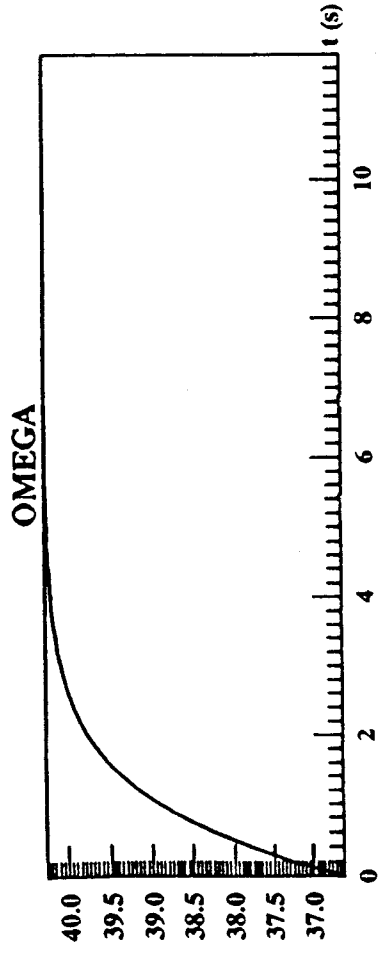


cas avec contrôle moteur

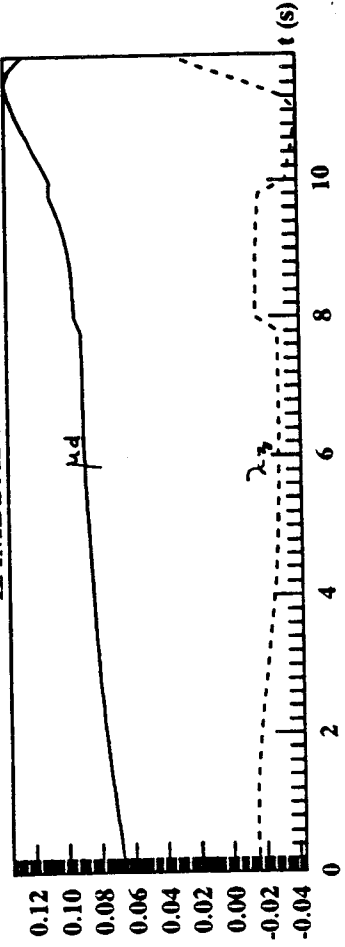
Résultat de 2 itérations par QSL

Initialisation avec les courbes précédentes

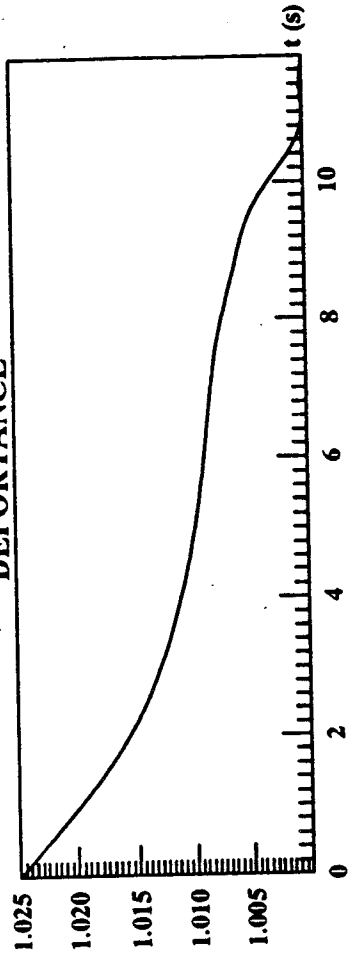




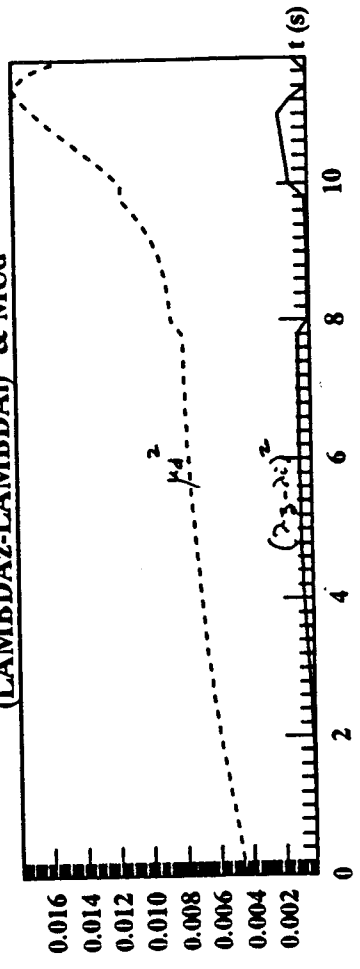
LAMBDAz & MUd



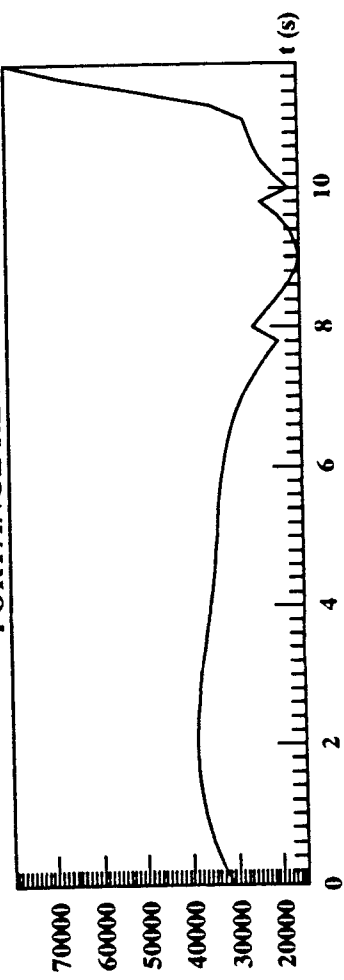
DEPORTANCE



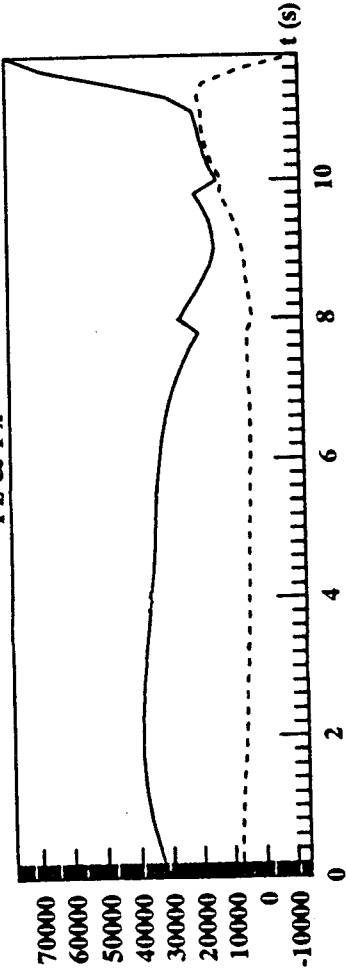
(LAMBDAz-LAMBDAj)^2 & MUD^2



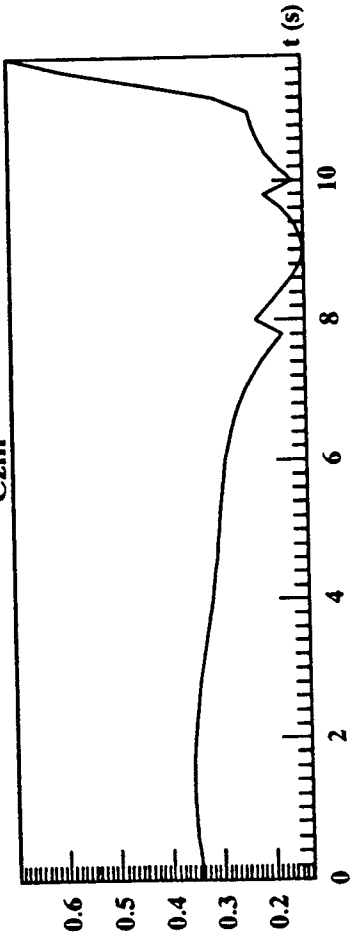
PORTANCE REELLE



Fz & Fx



Czm



REFERENCES BIBLIOGRAPHIQUES

Apkarian P., Champetier C., Magni J.F.(1990)

Design of a helicopter output feedback control law using modal and structures robustness techniques.

International Journal on control, à paraître.

Aumasson C., Landiech P. (1988)

Méthode itérative de type gradient projeté généralisé pour l'optimisation paramétrique et fonctionnelle de systèmes dynamiques soumis à des contraintes.

Note technique ONERA, 1988-9.

Belghith S., Rosset M.N. (1984)

Contrôle optimal: application des crochets de Lie à un problème de régulation thermique..

C.R. Académie des sciences Paris, t-299, série I, n°16, 1984, pp. 811-815.

Bellman R. (1957)

Dynamic programming.

Priceton University Press, N.J., 1957.

Bellman R., Kalaba R. (1965)

Dynamic programming and modern control theory .

Academic press NY 1965.

Betts J.T. (1988)

The application of sparse broyden updates in the collocation method for optimal control problems.
Proc. AIAA Guidance, Navigation and control conf.,Minneapolis 15-17 August 1988, pp. 814-820.

Betts J.T., Huffman W.P. (1990)

The application of sparse non linear programming to trajectory optimisation.

AIAA 90-3448-CP, pp. 1198-1213.

Blank D., Shinar J. (1978)

Two modified second order algorithms for the solution of constrained optimal control problems.
Technion Israel Institute of Technology, Department of Aeronautical Engineering, T.A.E. n° 309,
Aout 1978.

Blank D., Shinar J. (1978)

A direct shooting algorithm and the application of parametric optimization for constrained optimal control problem.
Technion Israel Institute of Technology, Department of Aeronautical Engineering, T.A.E. n° 311,
Aout 1978.

Blank D., Shinar J. (1982)

Efficient combinations of numerical algorithms for flight mechanics problems.
Journal of guidance, control and dynamics, vol.5, n°2, 1982, pp. 124-130.

Borne P., Dauphin-Tanguy G., Richard J.P., Rotella F., Zambettakis I. (1992)

Commande et optimisation des processus.
Méthodes et techniques de l'ingénieur, tome 1, ed. Technip, 1992.

Boudarel R., Delmas J., Guichet P. (1968)

Commande optimale des processus: programmation non lineaire et ses applications.
Techniques de l'automatisme, Dunod, Tome 2, 1968.

Bourdache-Siguerdidjane H., Fliess M. (1987)

Optimal feedback control of non-linear systems.
Automatica, vol.23, 1987, pp. 365-372.

Bourdache-Siguerdidjane H. (1987)

On application of a new method for computing optimal non-linear feedback controls
Optimal control applications & methods, vol.8, 1987, pp. 397-409.

Breakwell J.V., Speyer J.L., Bryson A.E. (1963)

Optimization and control of non linear systems using the second variation.
SIAM J.control, 1 1963, pp. 193-223..

Bryson A.E., Denham W.F., Dreyfus S.E. (1963)

Optimal programming problems with inequality constraints. Necessary conditions for external solutions.

A.I.A.A Journal, vol.1, n°11, 1963, pp. 2544-2550.

Bryson A.E., Desai M.N., Hoffman W.C. (1969)

Energy state approximation in performance optimization of supersonic aircraft.
Journal of aircraft, Novembre 1969, pp. 481-488.

Bryson A.E., Ho Y.C. (1975)

Applied Optimal Control.
J. Wiley & Sons, 1975.

Chen C.F., Hsiao C.H. (1975)

Walsh Series analysis in optimal control.
International Journal of Control, vol.21, n°6, 1975, pp. 881-897.

Chen W.L., Shih Y.P. (1978)

Analysis and optimal control of time varying linear system via walsh functions.
International Journal of Control, vol.27, n°6, 1978, pp. 917-932.

Choi C.H., Laub A.J. (1990)

Efficient matrix-valued algorithms for solving stiff riccati differential equations.
I.E.E.E. Trans. on automatic control, vol.35, n°7, 1990, pp. 770-776.

Claude D. (1984)

Linéarisation par difféomorphisme et immersion des systèmes.

Analysis and Optimization of Systems, Lecture Notes in control and information science, Springer Verlag, n°63, 1984, pp. 339-351.

Costes P. (1991)

Première formulation de la dynamique du vol des hélicoptères.

Rapport technique ONERA/DES 4/5161SY, Janvier 1991.

Dixon L.C.W., Bartholomew-biggs M.C (1981)

Adjoint control transformation for solving practical optimal control problems.

Optimal control applications and methods, John Wiley & sons vol.2, 1981, pp. 365-681.

Dyer P., Mc Reynolds S.R. (1970)

The computation and theory of optimal control .

Academic press NY 1970..

Farineau J.(1983)

Application de la théorie de la commande optimale au pilotage des hélicoptères.

Rapport Technique O.N.E.R.A./ C.E.R.T./ D.E.R.A. n°7274, Mars 1983.

Gill P.E., Murray W., Saunders M.A., Wright M.H. (1988)

Recent developpements in constrained optimization.

Journal of Computational and Applied Mathematics, vol 22, 1988, pp.257-270.

Garnot M.A. (1992)

Contribution to performance assesment of helicopter rotor speed control by numerical optimization.

Soumis pour publication, 18th European Rotorcraft Forum, 15-18 sept. 1992, Avignon.

Gimonet B., Labarrere M.(1980)

Etude des lois de pilotage d'un système stabilisé trois axes

Rapport Technique O.N.E.R.A./ C.E.R.T./ D.E.R.A. n°7242, Août 1980.

Goh C.J., Teo K.L. (1988)

Control parameterization : a unified approach to optimal control problems with general constraints.

Automatica, vol.24, n°1, 1988, pp. 3-18.

Goldstein A.A., Price J.F. (1967)

An effective algorithm for minimization.

Numerische mathematik, vol.10, n°1, 1967, pp.184-189.

Gottlieb R.G. (1967)

Rapid convergence to optimum solutions using a min-H strategy.

A.I.A.A. Journal, vol.5, n°2, 1967, pp. 322-329.

Grimm W., Berger E. (1985)

Benutzeranteilung für das Rechenprogramm BNDSCO zur Lösung beschränkter optimaler Steuerungsprobleme.

D.F.V.L.R. Mitt. 85-05

Hargraves C.R., Johnson F., Paris S.W., Rettie I. (1981)

Numerical computation of optimal atmospheric trajectories.

Journal of guidance and control, Aug. 1981, pp. 406-414.

Hargraves C.R., Paris S.W. (1987)

Direct trajectory optimization using nonlinear programming and collocation.

Journal of guidance, Aug. 1987, pp. 338-342.

Hestenes M. (1980)

Conjugate direction methods in optimization.

Application of mathematics, vol. 12, Springer verlag 1980.

Hougui S., Mishne D. (1990)

An approximate guidance law for atmospheric plane change maneuver of an aerocruise space vehicle.

A.I.A.A.-90-2832-CP.

Hull D.G., Edgeman J. (1975)

Suboptimal control using a second order parameter optimization method.

Journal of optimization theory and applications, vol.27, n°5/6, 1975, pp. 481-491.

Hull D.G., Williamson W.E. (1979)

Numerical derivatives for parameter optimization.

Journal of guidance and control, vol.2, n°2, 1979, pp. 158-160

Hull D.G. (1988)

On the variation process in parameter optimization.

Journal of optimization theory and applications, vol.56, n°1, 1988, pp. 31-38

Hull D.G. (1990)

On taking variation in optimal control theory.

A.I.A.A.- 90 -3487 - CP, 1990, pp. 1561-1567.

Huynh H. T. (1988)

Aircraft trajectories, computation-prediction-control.

Détermination des lois de guidage quasi-optimales en temps réel pour des trajectoires d'avions de combat.

AGARDograph n°301, vol.1

Hsu D.G., Cheng B. (1981)

Analysis and optimal control of time varying linear systems via blockpulse functions.
International journal of control, vol. 33, n°6, 1981, pp. 1107-1122.

Ilgenfritz D.H., Gailey J.W. (1988)

Trajectory optimization for proposed hypervelocity vehicles.
A.I.A.A. paper 88-4343 CP, 1988.

Isaev V.K., Sonin V.V. (1963)

On a modification of newton's method for the numerical solution of boundary value problems.
USSR computational mathematics and mathematical physics, 1963, pp. 1525-1528.

Jacobson D.H. (1968)

New second-order and first-order algorithms for determination of optimal control: a differential dynamic programming approach.
Journal of optimization theory and applications, vol.2, n°6, 1968, pp. 411-441.

Jacobson D.H., Mayne D.Q. (1970)

Differential dynamic programming
American Elsevier, New-York, 1970.

Johnson F.T. (1969)

Approximate finite thrust trajectory optimization.
A.I.A.A. Journal, June 1969, pp. 993-997.

Kalaba R. (1959)

On nonlinear differential equations, the maximum operation and monotone convergence.
Journal of mathematics and mechanics, vol.8, n°4, 1959, pp. 519-573.

Kalaba R., Bellman R. (1965)

Quasilinearization and non linear boundary value problems.

Modern analytic and computational methods in science and mathematics, vol.3, Elsevier NY 1965.

Kantorovich L.V. (1949)

On Newton's method.

Trudy math. inst. Steklov, 28, 1949, pp 104-144, English translation, Selected Articles on numerical analysis.

Keller H.B. (1976)

Numerical solution of two point boundary value problems.

SIAM ,n° 24, 1976.

Kelley H. J. (1964)

An optimal guidance approximation theory.

I.E.E.E. Trans. on automatic control, Octobre 1964, pp. 375-380.

Kelley H. J., Kopp R.E., Gardner Moyer H. (1964)

A trajectory optimization technique based upon the theory of the second variation.

Progress in astronautics and aeronautics, vol 14, Academic Press, N.Y.1964, pp. 559-582.

Kelly W.D. (1990)

A generalized version of two point boundary value problem guidance algorithm.

A.I.A.A. 90-3327 CD, 114-124.

Kuhn H.W., Tucker A.W.

Non linear programming.

Proc. of the 2° Berkeley symposium on mathematic, statistics and probability, University of California press.

Leondes C.T., paine G. (1968)

Extention in quasilinearization Techniques for optimal control.

Journal of optimization theory and applications, vol.2, n°5, 1968, pp.316-330.

Leondes C.T., paine G. (1968)

Computational results for extensions in quasilinearization techniques for optimal control.

Journal of optimization theory and applications, vol.2, n°6, 1968, pp.395-410.

Lewallen J.M. (1967)

A modified quasilinearization method for solving trajectory optimisation problems.

A.I.A.A Journal , vol.5, n° 5, 1967, pp. 962-965.

Lin Y.Y., Kraige L.G. (1989)

Enhanced techniques for solving the two point boundary value problem associated with the optimal attitude control of spacecraft.

The Journal of the Astronautical Sciences, vol.37, n°1, Jan.1989, pp 1-15.

Long R.S. (1965)

Quasilinearization and orbit determibation.

A.I.A.A. Journal, vol.3, n°10, 1965, pp. 1937-1940.

Luus R. (1974)

Optimal control by direct search on feedback gain matrix.

Chemical Engineering Science, 1974, vol.29, pp. 1013-1017.

Mc Cue G.A. (1967)

Quasilinearization determination of optimum finite thrust orbital transfer.

A.I.A.A. Journal, vol.5, n°4, 1967, pp. 755-763.

Mc Gill R., Kenneth P. (1963)

A convergence theorem on the iteration solution of non linear two point boundary solution.
Proc. 14° Conf Astro., vol.4 Gauthier-Villars Paris 1963, pp. 173-188.

Mc Gill R., Kenneth P. (1964)

Solution of variationnal problems by mean of a generalized Newton-Raphson Operators.
A.I.A.A. Journal, vol.2, n°10, 1964, pp. 1761-1766.

Marchal C., Contensou P. (1981)

Singularities in optimization of deterministic dynamic systems.
A.I.A.A. 81-4121 , pp. 240-252.

Menon P.K.A., Cheng V.H.L., Lin C.A., Briggs M.M. (1987)

High-performance missile synthesis with trajectory and propulsion system optimization.
Journal Spacecraft, vol.24, n°6, Nov-Dec 1987, pp. 552-557.

Miele A., Damoulakis J.N., Cloutier J.R., Tietze J.L. (1974)

Sequential gradient restoration algorithm for optimal control problems with non differential constraints.
Journal of optimization theory and applications, vol.13, n°2, 1974, pp. 218-255.

Miele A., Mangiavacchi A., Aggarwal A.K. (1974)

Modified quasilinearisation algorithm for optimal control problems with non differential constraints.
Journal of optimization theory and applications, vol.14, n°5, 1974, pp. 529-555.

Mitter S.K. (1966)

Successive approximation methods for the solution of optimal control problems.
Automatica, vol.3 ,1966, pp. 135-149.

Moreigne O. (1984)

Contribution à la modélisation et à la synthèse des systèmes à échelles de temps multiples. Application au calcul en temps réel d'une commande quasi-optimale pour les trajectoires d'avion de combat.

thèse de docteur-ingénieur I.D.N. (1984).

Murray W., Gill P.E., Saunders M.A., Wright M.H. (1988)

Recent developments in constrained optimization.

Journal of computational and applied mathematics, n°22, 1988.

Paraskevopoulos P.N. (1983)

Chebyshev series approach to system identification, analysis and optimal control.

Journal of Franklin institute, vol. 316, Aug. 1983, pp. 135-157.

Pesch H.J. (1989)

Real time computation of feedback controls for constrained optimal control problems , part 1: neighboring extremals .

Optimal control applications and methods, vol.10, 1989, pp. 139-145.

Pesch H.J. (1989)

Real time computation of feedback controls for constrained optimal control problems, part 2 : a correction method based on multiple shooting..

Optimal control applications and methods, vol.10, 1989, pp. 147-171.

Peterson S.R., Rader J.E. (1981)

time-constrained maximum energy turn.

A.I.A.A. paper 81-1886 ,Atmospheric flight mechanics conf., Aug. 19-21 1981 Albuquerque.

Pontryagin L.S., Boltyanskii V.G., Gamkrelidze R.V., Mishchenko E.F. (1962)

The mathematical theory of optimal processes.

Kohn Wiley & sons, Interscience publishers, N.Y. 1962.

Rader J.E. , Hull D.G. (1975)

Computation of optimal aircraft trajectories using parameter optimization methods.
Journal of aircraft, vol.12, Nov. 1975, pp. 864-866.

Reboulet C., Champetier C. (1984)

A new method for linearizing non linear systems : the pseudolinearization.
International Journal on Control, vol.40, n°4, 1984, pp. 631-638.

Reboulet C., Mouyon P., Champetier C.(1986)

About the local linearization of on linear systems.
Algebraic and geometric methods in non linear control theory, 1986, pp. 311-322.

Rotella F. (1991)

Derivatives of matrices.
J. of F. I. , vol 328, pp.487 ,1991.

Robinson S.M. (1971)

Extension of Newton's method to mixed systems of non linear equations and inequalities.
Tech. Summ. Rep 1161, Rep 1194, Mathematics Research center, Univ. of Wisconsin, Madison,
1971.

Rotella F., Dauphin-Tanguy G (1988)

Non-linear systems: identification and optimal control.
International Journal of Control, vol.48, n°2, 1988, pp. 525-544.

Rusnak I , Meir L.(1988)

Optimal guidance for high order and acceleration constrained missile.
ICAS 88-1.3.2, pp. 383-391.

Rusnak I (1988)

Almost analytic representation for the solution of the differential matrix riccati equation.
I.E.E.E. trans. on automatic control, vol.33, n°2, 1988, pp. 191-193.

Russel R.D., Shampine L.F.(1972)

A collocation method for boundary value problems.
Numerische mathematik, n°19, 1972, pp. 1-28.

Sheela B.V., Ramamoorthy P. (1981)

Optimal control via mathematical programming.
Journal of guidance and control, Aug. 1981, pp. 443-444.

Schaefer C.G., Lutze F (1990)

Enhanced energy maeuverability for attack helicopters using continuous , variable (C-V) rotor speed control.
47th American Helicopter Society Forum 6-8 May 1991, Phenix, Arizona.

Sirisena H.R. (1973)

Computation of optimal Control using a piecewise polynomial parameterization.
I.E.E.E. Trans. on auto. control, vol. A.C.18, Aug. 1973, pp. 409-411.

Sirisena H.R., Tan K.S.(1974)

Computation of constrained optimal Control using parameterization technics.
I.E.E.E. Trans. on auto. control, vol. A.C.19, Aug. 1974, pp. 431-433.

Sirisena H.R., Chou F.S.(1976)

An efficient algorithm for solving optimal control problems with linear constraints.
I.E.E.E. Trans. on auto. control, vol. A.C.21, April 1976, pp. 275-277.

Stavroulakis P., Tzafestas S. (1977)

Walsh Series approach to observer filter design in optimal control systems.
International Journal of Control, vol.26, n°5, 1977, pp. 721-736.

Talwar D., Sivan R. (1989)

An efficient Numerical algorithm for the solution of a class of optimal control problems.
I.E.E.E. Trans. on automatic control, vol. 34, n°12, Dec.1989, pp. 1308-1311.

Tapley B.D., Lewallen J.M.(1967)

Comparison of several numerical optimization methods.
Journal of optimization theory and applications, vol.1, n°1, 1967, pp. 1-31.

Vlassenbroek J., Van Doren R. (1988)

A Chebychev technique for solving non linear optimal control problems.
I.E.E.E. Trans. on auto. control, vol. A.C.33, April 1988, pp. 333-340.

Weston A., Cliff G., Kelley H (1985)

On board near optimal climb dash energy management..
Journal of Guidance, vol. 8, n°3, 1985, pp. 320-324.

Williamson W.E. (1971)

Use of polynomial approximation to calculate suboptimal controls.
A.I.A.A. Journal, Nov. 1971, pp. 2271-2273.

Willemstein A.P. (1975)

Optimal regulation of non linear dynamical systems on a finite interval.
SIAM Journal of control and optimization , vol.15, Nov. 1975, pp. 1050-1069

Yeo B.P., Goh B.S., Waldron J. (1974)

Optimal initial choice of multipliers in the quasilinearization method for optimal control problems with boundary controls.

International Journal of control, vol.20, n°1, 1974, pp. 17-33.

Yeo B.P.(1974)

A quasilinearization algorithm and its application to a manipulator problem.

International Journal of control, vol.20, n°4, 1974, pp. 623-640.

Yeo B.P.(1976)

An extension in optimal initial choice of multipliers in the quasilinearization.

International Journal of control, vol.24, n°5, 1976, pp. 593-608.

Yeo B.P.(1976)

Quasilinearization and optimal control problems with control bounds.

A.I.A.A. Journal, vol.14, n°7, 1976, pp. 963-966.

Yeo B.P., Sng K.B.(1980)

Numerical solution of the constrained reentry vehicle trajectory problem via quasilinearisation.

Journal of guidance and control, vol.3, n°5, 1980, pp. 392-397.

ANNEXES

annexe 1 : Variation de la condition de transversalité

Considérons ω

$$\omega = H(\vec{x}(t), \vec{u}(t), \vec{\lambda}(t), t) + \frac{\partial \Phi}{\partial t}(\vec{x}(t), t, \vec{v}) \Big|_{t_f} \quad (\text{A1.1})$$

Soit les différentielles

$$d\vec{x}(t_f) = \delta\vec{x}(t_f) + \dot{\vec{x}}(t_f) dt_f \quad (\text{A1.2})$$

$$d\vec{\lambda}(t_f) = \delta\vec{\lambda}(t_f) + \dot{\vec{\lambda}}(t_f) dt_f \quad (\text{A1.3})$$

Supposons que L et f ne dépendent pas explicitement du temps .Alors on a

$$\begin{aligned} d\omega &= d\left(H(\vec{x}(t), \vec{u}(t), \vec{\lambda}(t), t) + \frac{\partial \Phi}{\partial t}(\vec{x}(t), t, \vec{v}) \Big|_{t_f} \right) \\ &= \left(\frac{\partial H}{\partial \vec{x}} + \frac{\partial^2 \Phi}{\partial t \partial \vec{x}} \right) . d\vec{x}_f + \vec{f}^T . d\vec{\lambda}_f + \frac{\partial \vec{\psi}^T}{\partial t} . d\vec{v} + \frac{\partial^2 \Phi}{\partial t^2} . dt_f + \frac{\partial H}{\partial \vec{u}} . d\vec{u}_f \\ &= \left(\frac{\partial H}{\partial \vec{x}} + \frac{\partial^2 \Phi}{\partial t \partial \vec{x}} \right) \delta\vec{x}_f + \vec{f}^T . \delta\vec{\lambda}_f + \frac{\partial \vec{\psi}^T}{\partial t} . \delta\vec{v} \\ &\quad + \left(\frac{\partial H}{\partial \vec{x}} + \frac{\partial^2 \Phi}{\partial t \partial \vec{x}} \right) \dot{\vec{x}}_f + \vec{f}^T . \dot{\vec{\lambda}}_f + \frac{\partial^2 \Phi}{\partial t^2} . dt_f + \frac{\partial H}{\partial \vec{u}} . d\vec{u}_f \quad (\text{A1.4}) \end{aligned}$$

comme $\vec{u}^* = \text{Argmin } H$ on a $\frac{\partial H}{\partial \vec{u}} = 0$ et

$$d\omega = \left(\frac{\partial H}{\partial \vec{x}} + \frac{\partial^2 \Phi}{\partial t \partial \vec{x}} \right) \delta\vec{x}_f + \vec{f}^T . \delta\vec{\lambda}_f + \frac{\partial \vec{\psi}^T}{\partial t} . \delta\vec{v}$$

$$+ \left(\left(\frac{\partial H}{\partial \vec{x}} + \frac{\partial^2 \Phi}{\partial t \partial \vec{x}} \right) \dot{\vec{x}}_f + \vec{f}^T \cdot \dot{\vec{\lambda}}_f + \frac{\partial^2 \Phi}{\partial t^2} \right) dt_f \quad (\text{A1.5})$$

Posons

$$d \left(\vec{\lambda}_f^T - \frac{\partial \Phi}{\partial \vec{x}} \right) = - \left(\vec{\lambda}_f^T - \frac{\partial \Phi}{\partial \vec{x}} \right) \quad (\text{A1.6})$$

nous avons alors

$$d \vec{\lambda}_f = \frac{\partial^2 \Phi}{\partial \vec{x}^2} \cdot d \vec{x}_f + \frac{\partial \vec{\psi}^T}{\partial t} \cdot d \vec{v} + \frac{\partial^2 \Phi^T}{\partial \vec{x} \partial t} \cdot dt_f - \left(\vec{\lambda}_f^T - \frac{\partial \Phi}{\partial \vec{x}} \right) \quad (\text{A1.7})$$

$$\begin{aligned} \delta \vec{\lambda}_f &= \frac{\partial^2 \Phi}{\partial \vec{x}^2} \cdot \delta \vec{x}_f + \frac{\partial \vec{\psi}^T}{\partial t} \cdot \delta \vec{v} + \left(\frac{\partial^2 \Phi^T}{\partial \vec{x} \partial t} - \dot{\vec{\lambda}}_f + \frac{\partial^2 \Phi}{\partial \vec{x}^2} \cdot \dot{\vec{x}}_f \right) dt_f \\ &\quad - \left(\vec{\lambda}_f^T - \frac{\partial \Phi}{\partial \vec{x}} \right) \end{aligned} \quad (\text{A1.8})$$

et

$$\begin{aligned} d\omega &= - \vec{f}^T \cdot \left(\vec{\lambda}_f - \frac{\partial \Phi^T}{\partial \vec{x}} \right) \\ &\quad + \left(\frac{\partial H}{\partial \vec{x}} + \frac{\partial^2 \Phi}{\partial \vec{x} \partial t} + \vec{f}^T \cdot \frac{\partial^2 \Phi}{\partial \vec{x}^2} \right) \delta \vec{x}_f + \left(\frac{\partial \vec{\psi}}{\partial t} + \frac{\partial \vec{\psi}}{\partial \vec{x}} \vec{f} \right)^T \cdot \delta \vec{v} \\ &\quad + \left(\left(\frac{\partial H}{\partial \vec{x}} + \frac{\partial^2 \Phi}{\partial \vec{x} \partial t} \right) \dot{\vec{x}}_f + \vec{f}^T \cdot \dot{\vec{\lambda}}_f + \frac{\partial^2 \Phi}{\partial t^2} \right) \cdot dt_f \\ &\quad + \vec{f}^T \cdot \left(\frac{\partial^2 \Phi^T}{\partial \vec{x} \partial t} - \dot{\vec{\lambda}}_f + \frac{\partial^2 \Phi}{\partial \vec{x}^2} \cdot \dot{\vec{x}}_f \right) \cdot dt_f \end{aligned} \quad (\text{A1.9})$$

soit

$$\begin{aligned} d\omega &= - \vec{f}^T \cdot \left(\vec{\lambda}_f - \frac{\partial \Phi^T}{\partial \vec{x}} \right) \\ &\quad + \left(\frac{\partial H}{\partial \vec{x}} + \frac{\partial^2 \Phi}{\partial \vec{x} \partial t} + \vec{f}^T \cdot \frac{\partial^2 \Phi}{\partial \vec{x}^2} \right) \delta \vec{x}_f + \left(\frac{\partial \vec{\psi}}{\partial t} + \frac{\partial \vec{\psi}}{\partial \vec{x}} \vec{f} \right)^T \cdot \delta \vec{v} \\ &\quad \dots \left(\vec{f}^T \cdot \left(\frac{\partial^2 \Phi^T}{\partial \vec{x} \partial t} + \frac{\partial^2 \Phi}{\partial \vec{x}^2} \cdot \dot{\vec{x}}_f \right) + \left(\frac{\partial H}{\partial \vec{x}} + \frac{\partial^2 \Phi}{\partial \vec{x} \partial t} \right) \dot{\vec{x}}_f + \frac{\partial^2 \Phi}{\partial t^2} \right) \cdot dt_f \end{aligned}$$

(A1.10)

Cas particuliers :

1/ Si la dynamique est vérifiée, alors

$$\begin{aligned}\dot{\vec{x}} &= \vec{f}^*(\vec{x}, \vec{\lambda}, t) \\ \dot{\vec{\lambda}} &= \vec{h}^*(\vec{x}, \vec{\lambda}, t) \\ \left(\frac{\partial H}{\partial \vec{x}} + \frac{\partial^2 \Phi}{\partial \vec{x} \partial t} \right) + \vec{f}^T \cdot \frac{\partial^2 \Phi}{\partial \vec{x}^2} &= \frac{d}{dt} \left(\frac{\partial \Phi^T}{\partial \vec{x}} \right) - \dot{\vec{\lambda}} \\ \left(\frac{\partial \vec{\psi}}{\partial t} + \frac{\partial \vec{\psi}}{\partial \vec{x}} \vec{f} \right) &= \frac{d\vec{\psi}}{dt}\end{aligned}$$

et

$$\begin{aligned}d\omega &= - \vec{f}^T \cdot \left(\vec{\lambda}_f - \frac{\partial \Phi^T}{\partial \vec{x}} \right) \\ &+ \left(\frac{d}{dt} \left(\frac{\partial \Phi^T}{\partial \vec{x}} \right) - \dot{\vec{\lambda}}_f \right) \cdot \delta \vec{x}_f + \frac{d\vec{\psi}^T}{dt} \cdot \delta \vec{v} \\ &+ \left(\vec{f}^T \cdot \frac{\partial^2 \Phi^T}{\partial \vec{x} \partial t} + \left(\frac{d}{dt} \left(\frac{\partial \Phi^T}{\partial \vec{x}} \right) - \dot{\vec{\lambda}}_f^T \right) \cdot \dot{\vec{x}}_f + \frac{\partial^2 \Phi}{\partial t^2} \right) \cdot dt_f \quad (A1.11)\end{aligned}$$

2/ Si de plus les conditions finales sur l'adjoint sont vérifiées

$$\vec{\lambda}^T(t_f) = \frac{\partial \Phi}{\partial \vec{x}} (\vec{x}(t), t, \vec{v}) \Big|_{t_f}$$

alors

$$\frac{d}{dt} \left(\frac{\partial \Phi^T}{\partial \vec{x}} \right) - \dot{\vec{\lambda}}_f = \frac{\partial \omega^T}{\partial \vec{x}} \quad (A1.12)$$

et

$$\begin{aligned}d\omega &= - \vec{f}^T \cdot \left(\vec{\lambda}_f - \frac{\partial \Phi^T}{\partial \vec{x}} \right) \\ &+ \frac{\partial \omega^T}{\partial \vec{x}} \cdot \delta \vec{x}_f + \frac{d\vec{\psi}^T}{dt} \cdot \delta \vec{v}\end{aligned}$$

$$+ \left(\vec{f}^T \cdot \frac{\partial^2 \Phi^T}{\partial \vec{x} \partial t} + \frac{\partial \omega \cdot \dot{\vec{x}}}{\partial \vec{x}} \cdot \vec{x}_f + \frac{\partial^2 \Phi}{\partial t^2} \right) \cdot dt_f \quad (\text{A1.13})$$

en effet

$$\begin{aligned} \frac{d}{dt} \left(\frac{\partial \Phi^T}{\partial \vec{x}} \right) \cdot \dot{\vec{\lambda}} &= \frac{\partial}{\partial t} \left(\frac{\partial \Phi^T}{\partial \vec{x}} \right) + \frac{\partial}{\partial \vec{x}} \left(\frac{\partial \Phi^T}{\partial \vec{x}} \right) \cdot \dot{\vec{x}} + \left(\frac{\partial L^T}{\partial \vec{x}} + \frac{\partial f^T}{\partial \vec{x}} \cdot \dot{\vec{\lambda}} \right) \\ &= \frac{\partial}{\partial t} \left(\frac{\partial \Phi^T}{\partial \vec{x}} \right) + \frac{\partial}{\partial \vec{x}} \left(\frac{\partial \Phi^T}{\partial \vec{x}} \right) \cdot \vec{f} + \frac{\partial L^T}{\partial \vec{x}} + \frac{\partial f^T}{\partial \vec{x}} \cdot \frac{\partial \Phi^T}{\partial \vec{x}} \\ &= \frac{\partial}{\partial \vec{x}} \left(\frac{\partial \Phi}{\partial t} + \frac{\partial f^T}{\partial \vec{x}} \cdot \frac{\partial \Phi^T}{\partial \vec{x}} + L \right)^T \end{aligned} \quad (\text{A1.14})$$

annexe 2 : solutions analytique du problème linéaire quadratique.

La commande optimale est donnée par

$$u(t) = -\lambda_v(t)$$

Pour le cas où la commande n'est pas bornée, la solution est immédiate. Pour le cas où la commande est limitée par u_{\min} ou u_{\max} , nous aurons une ou deux commutations selon la valeur des bornes (si les contraintes deviennent actives pendant l'intervalle de temps).

1) cas sans contrainte

$$u(t) = v_x \cdot t - (v_x t_f + v_v) \quad (A2.1)$$

$$x(t) = \frac{v_x}{6} \cdot t^3 - (v_x t_f + v_v) \cdot \frac{t^2}{2} + v_o \cdot t + x_o \quad (A2.2)$$

$$v(t) = \frac{v_x}{2} \cdot t^2 - (v_x t_f + v_v) \cdot t + v_o \quad (A2.3)$$

$$\lambda_x(t) = v_x \quad (A2.4)$$

$$\lambda_v(t) = -v_x \cdot t + (v_x t_f + v_v) \quad (A2.5)$$

$$v_x = \frac{12}{t_f} \cdot \left[(x_o - x_f) + \frac{t_f}{2} \cdot (v_o + v_f) \right] \quad (A2.6)$$

$$v_v = \frac{6}{t_f} \cdot \left[(x_f - x_o) - \frac{t_f}{3} \cdot (v_o + 2v_f) \right] \quad (A2.7)$$

2) une seule contrainte : $u > u_{\max}$

Soit t_c l'instant de commutation. Pour le cas où $x_o = 0$ et $v_o = 0$, l'état et la commande seront

pour $t \leq t_c$

$$u(t) = u_{\max} \quad (A2.8)$$

$$x(t) = u_{\max} \frac{t^2}{2} \quad (A2.9)$$

$$v(t) = u_{\max} t \quad (A2.10)$$

pour $t \geq t_c$

$$u(t) = v_x (t - t_f) - v_v \quad (\text{A2.11})$$

$$x(t) = \frac{v_x}{6} (t - t_f)^3 - \frac{v_v}{2} (t - t_f)^2 + v_f (t - t_f) + x_f \quad (\text{A2.12})$$

$$v(t) = \frac{v_x}{2} (t - t_f)^2 - v_v (t - t_f) + v_f \quad (\text{A2.13})$$

avec

$$t_c = \frac{3x_f + v_f \cdot t_f + u_{\max} \frac{t_f^2}{2}}{u_{\max} t_f - v_f}$$

$$v_x = \frac{2 (v_f - u_{\max} t_f)}{(t_c - t_f)^2} \quad (\text{A2.14})$$

$$v_v = v_x(t_c - t_f) - u_{\max} \quad (\text{A2.15})$$

3) une seule contrainte : $u < u_{\min}$

Soit t_c l'instant de commutation. Pour le cas où $x_0 = 0$ et $v_0 = 0$ l'état et la commande seront

pour $t \leq t_c$

$$u(t) = v_x t - (v_x t_f + v_v) \quad (\text{A2.16})$$

$$x(t) = \frac{v_x}{6} t^3 - \frac{(v_x t_f + v_v)}{2} t^2 \quad (\text{A2.17})$$

$$v(t) = \frac{v_x}{2} t^2 - (v_x t_f + v_v) t \quad (\text{A2.18})$$

et pour $t \geq t_c$

$$u(t) = u_{\min} \quad (\text{A2.19})$$

$$x(t) = u_{\min} \frac{(t - t_f)^2}{2} + v_f (t - t_f) + x_f \quad (\text{A2.20})$$

$$v(t) = u_{\min} (t - t_f) + v_f \quad (\text{A2.21})$$

avec

$$t_c = \frac{3x_f - 3v_f \cdot t_f + 3u_{\min} \frac{t_f^2}{2}}{u_{\min} t_f - v_f}$$

$$v_x = \frac{2 (u_{\min} t_f - v_f)}{t_c^2} \quad (\text{A2.22})$$

$$v_v = v_x(t_c - t_f) - u_{\min} \quad (\text{A2.23})$$

4) cas avec deux bornes

Soient t_1 et t_2 les instants de commutation entre u_{\max} , $u^*(t)$ et u_{\min} . Les résultats sont donnés pour $x_0 = 0$ et $v_0 = 0$. La continuité de la commande donne les relations :

$$t_1 = t_f + \frac{v_v + u_{\max}}{v_x} \quad (\text{A2.24})$$

$$t_2 = t_f + \frac{v_v + u_{\min}}{v_x} \quad (\text{A2.25})$$

Les adjoints seront

$$\lambda_x(t) = v_x \quad (\text{A2.26})$$

$$\lambda_v(t) = -v_x \cdot t + (v_x t_f + v_v) \quad (\text{A2.27})$$

L'état et la commande seront pour $t \leq t_1$

$$u(t) = u_{\max} \quad (\text{A2.28})$$

$$x(t) = u_{\max} \frac{t^2}{2} \quad (\text{A2.29})$$

$$v(t) = u_{\max} \cdot t \quad (\text{A2.30})$$

pour t tel que $t_1 \leq t \leq t_2$

$$u(t) = -\lambda_v \quad (\text{A2.32})$$

$$x(t) = v_x \frac{(t-t_1)^3}{6} - u_{\max} \frac{(t-t_1)^2}{2} + v_1 (t-t_1) + x_1 \quad (\text{A2.33})$$

$$v(t) = v_x \frac{(t-t_1)^2}{2} - u_{\max}(t-t_1) + v_1 \quad (\text{A2.34})$$

avec $v_1 = u_{\max} t_1 \quad (\text{A2.35})$

$$x_1 = u_{\max} \frac{t_1^2}{2} \quad (\text{A2.36})$$

et pour t tel que $t \geq t_2$

$$u(t) = u_{\min} \quad (\text{A2.37})$$

$$x(t) = u_{\min} \frac{(t-t_f)^2}{2} + v_f(t-t_f) + x_f$$

$$v(t) = u_{\min}(t-t_f) + v_f \quad (\text{A2.38})$$

En utilisant les équations de continuité en t_1 et t_2 , nous obtenons les relations qui permettent de calculer v_x , v_v , t_1 et t_2

$$u_{\max} = -v_x \cdot t_1 + (v_x t_f + v_v)$$

$$v_x(t_2 - t_1) + u_{\max} = u_{\min}$$

$$v_x \frac{(t_2 - t_1)^3}{6} - u_{\max} \frac{(t_2 - t_1)^2}{2} + u_{\max} t_1 (t_2 - t_1) + u_{\max} \frac{t_1^2}{2} = u_{\min} \frac{(t_2 - t_f)^2}{2} +$$

$$v_f(t_2 - t_f) + x_f$$

$$v_x \frac{(t_2 - t_1)^2}{2} - u_{\max}(t_2 - t_1) + u_{\max} t_1 = u_{\min}(t_2 - t_f) + v_f$$

**Annexe 3 : Matrices de changement de repère
application du théorème de la dynamique**

Les matrices de changement de repères sont notées **T** entre le repère aérodynamique et le repère avion, **R** entre le repère terrestre et le repère avion et **R'** entre le repère terrestre et le repère aérodynamique:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \mathbf{T}(\alpha, \beta) \cdot \begin{pmatrix} x_a \\ y_a \\ z_a \end{pmatrix}$$

$$\mathbf{T}(\alpha, \beta) = \begin{pmatrix} \cos\alpha\cos\beta & -\cos\alpha\sin\beta & -\sin\alpha \\ \sin\beta & \cos\beta & 0 \\ \sin\alpha\cos\beta & -\sin\alpha\sin\beta & \cos\alpha \end{pmatrix}$$

où α est l'angle d'incidence
 β l'angle de dérapage

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \mathbf{R}(\psi, \theta, \Phi) \cdot \begin{pmatrix} x_o \\ y_o \\ z_o \end{pmatrix}$$

$$\mathbf{R}(\psi, \theta, \Phi) = \begin{pmatrix} \cos\psi\cos\theta & \sin\psi\cos\theta & -\sin\theta \\ \sin\psi\cos\theta + \cos\psi\sin\theta\sin\Phi & \cos\psi\cos\Phi + \sin\psi\sin\theta\sin\Phi & \cos\theta\sin\Phi \\ \sin\psi\sin\Phi + \cos\psi\sin\theta\cos\Phi & \cos\psi\sin\Phi + \sin\psi\sin\theta\cos\Phi & \cos\theta\cos\Phi \end{pmatrix}$$

où ψ est l'azimuth
 θ l'assiette longitudinale
 Φ l'angle de gîte

$$\begin{pmatrix} x_a \\ y_a \\ z_a \end{pmatrix} = \mathbf{R}'(\chi, \gamma, \mu) \cdot \begin{pmatrix} x_o \\ y_o \\ z_o \end{pmatrix}$$

$$\mathbf{R}'(\psi, \theta, \Phi) = \begin{pmatrix} \cos\chi\cos\gamma & \sin\chi\cos\gamma & -\sin\gamma \\ -\sin\chi\cos\gamma + \cos\chi\sin\gamma\sin\mu & \cos\chi\cos\gamma + \sin\chi\sin\gamma\sin\mu & \cos\gamma\sin\mu \\ \sin\chi\sin\gamma + \cos\chi\sin\gamma\cos\mu & -\cos\chi\sin\gamma + \sin\chi\sin\gamma\cos\mu & \cos\gamma\cos\mu \end{pmatrix}$$

où χ, γ, μ sont les angles d'Euler

Dans le repère aérodynamique, le théorème de la dynamique donne

$$\left(\frac{d\vec{V}}{dt}\right)_{R_a} + \vec{\Omega}_{R_a} \wedge \vec{V} = \frac{\vec{F}}{M} + \frac{\vec{R}}{M} + \vec{g}$$

où $\left(\frac{d\vec{V}}{dt}\right)_{R_a}$ est la dérivée relative de \vec{V} dans R_a

et $\vec{\Omega}_{R_a}$ le vecteur instantané de rotation.

$$\left(\frac{d\vec{V}}{dt}\right)_{R_a} = \begin{pmatrix} \dot{V} \\ 0 \\ 0 \end{pmatrix}$$

$$\vec{\Omega}_{R_a} = \begin{pmatrix} -\dot{\chi} \sin \gamma \\ \dot{\gamma} \\ \dot{\chi} \cos \gamma \end{pmatrix}$$

$$\vec{g} = \begin{pmatrix} -g \sin \gamma \\ 0 \\ g \cos \gamma \end{pmatrix}$$

ce qui donne

$$\begin{aligned} \dot{V} &= \frac{1}{M} (R_{xa} + F_{xa}) - g \sin \gamma \\ 0 + V \dot{\chi} \cos \gamma &= \frac{1}{M} (R_{ya} + F_{ya}) - \frac{1}{M} (R_{za} + F_{za}) \sin \mu \\ 0 - V \dot{\chi} &= \frac{1}{M} (R_{za} + F_{za}) \cos \mu + g \cos \gamma \end{aligned}$$

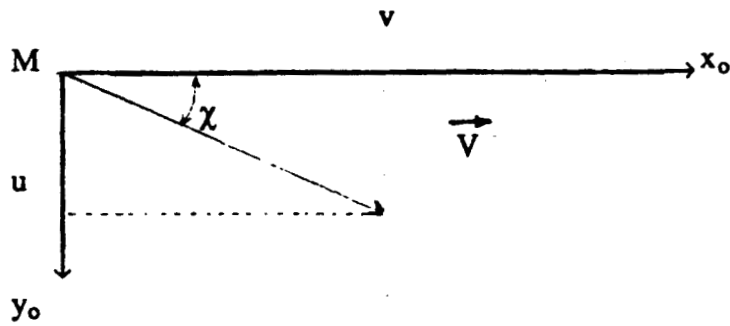
Annexe 4 : changement de variables

Soit le système

$$\begin{aligned}\dot{x} &= V \cos\chi \\ \dot{y} &= V \sin\chi \\ \dot{V} &= \frac{1}{M} (F - R_x) \\ \dot{\chi} &= \frac{g}{V} (n_z \sin\mu) \text{ avec } n_z = 1/\cos\mu\end{aligned}$$

Le changement de variable est le suivant

$$\begin{aligned}u &= V \cos\chi \\ v &= V \sin\chi\end{aligned}$$

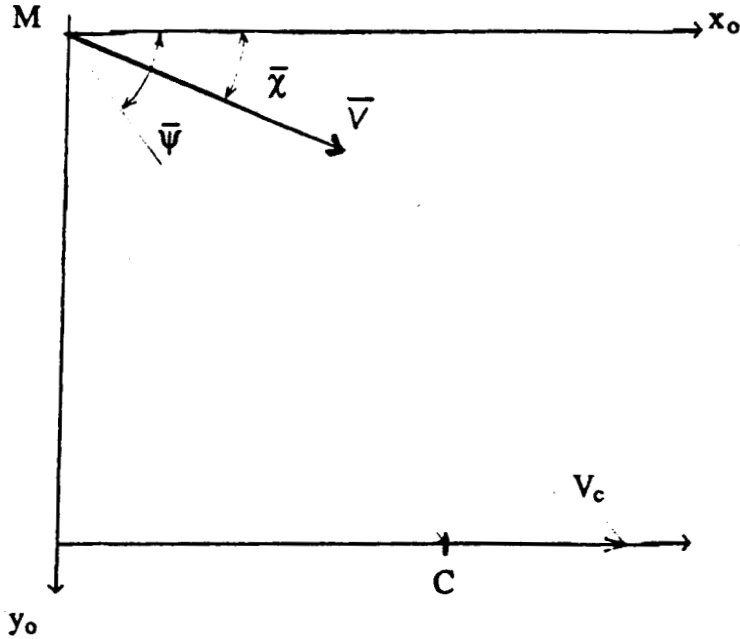


ce qui donne

$$\begin{aligned}\dot{u} &= \dot{V} \cos\chi - V \sin\chi \dot{\chi} \\ &= \frac{g}{V} \left(\frac{F}{Mg} - \frac{R_x}{Mg} \right) V \cos\chi - V \sin\chi \left(\frac{g}{V} \cdot n_z \sin\mu \right) \\ &= \frac{g}{V} \left\{ (F_m - (D_0 + D_1 n_z^2)) \cdot u - v \cdot \sqrt{n_z^2 - 1} \right\} \\ \dot{v} &= \dot{V} \sin\chi + V \cos\chi \dot{\chi} \\ &= \frac{g}{V} \left\{ (F_m - (D_0 + D_1 n_z^2)) \cdot v + u \cdot \sqrt{n_z^2 - 1} \right\}\end{aligned}$$

Annexe 5 : commande par perturbations singulières

Dans le cas d'une interception horizontale, la solution intérieure sera donnée par une trajectoire de collision (cf figure).



$$F(\bar{V}) - D_0(\bar{V}) - D_1(\bar{V}) = 0$$

$$V \sin \bar{\chi} = \bar{V} \sin(\bar{\psi} - \bar{\chi})$$

$$\tan \bar{\psi} = \frac{y_c - y_0}{x_c - x_0}$$

A chaque instant, on calcule μ_{marge} tel que

$$\tan \mu_{\text{marge}} = \frac{F - D_0 - D_1}{D_1} \quad (\text{fonctions de la vitesse courante})$$

La commande μ par perturbation singulière sera alors telle que

$$\tan \mu_{\text{pert}} = \sqrt{\frac{2V}{\bar{V} - V}} \cdot \sin \frac{(\bar{\chi} - \chi)}{2} \cdot \tan \mu_{\text{marge}}$$

Annexe 7 : calcul des dérivées partielles du système hélicoptère

$$F_n = \frac{1}{6} \rho S \sigma (\Omega R)^2 C_{2\alpha} \left(\frac{\theta_0 + \frac{3}{2} \lambda_z}{1 + \frac{\sigma C_{2\alpha}}{8 \mu_d}} \right)$$

$$\frac{\partial F_n}{\partial \theta_0} = \frac{1}{6} \rho S \sigma (\Omega R)^2 C_{2\alpha} \cdot \frac{1}{1 + \frac{\sigma C_{2\alpha}}{8 \mu_d}}$$

$$\begin{aligned} \frac{\partial F_n}{\partial \lambda_z} &= \frac{1}{6} \rho S \sigma (\Omega R)^2 C_{2\alpha} \cdot \left[\frac{\frac{3}{2} \mu_d \left(1 + \frac{\sigma C_{2\alpha}}{8 \mu_d}\right) - \left(\theta_0 + \frac{3}{2} \lambda_z\right) \frac{\sigma C_{2\alpha}}{8 \mu_d} \cdot \frac{\lambda_z}{\mu_d}}{\left(1 + \frac{\sigma C_{2\alpha}}{8 \mu_d}\right)^2} \right] \cdot \frac{1}{\left(1 + \frac{\sigma C_{2\alpha}}{8 \mu_d}\right)} \\ &= \frac{1}{6} \rho S \sigma (\Omega R)^2 C_{2\alpha} \cdot \left[\frac{\frac{3}{2} \mu_d}{\left(1 + \frac{\sigma C_{2\alpha}}{8 \mu_d}\right)} - \left(\theta_0 + \frac{3}{2} \lambda_z\right) \cdot \frac{\frac{\sigma C_{2\alpha}}{8 \mu_d} \cdot \frac{\lambda_z}{\mu_d}}{\left(1 + \frac{\sigma C_{2\alpha}}{8 \mu_d}\right)^2} \right] \end{aligned}$$

$$\begin{aligned} \frac{\partial F_n}{\partial V_x} &= \frac{1}{6} \rho S \sigma (\Omega R)^2 C_{2\alpha} \cdot \frac{V_x}{V^2} \cdot \left[\frac{\frac{3}{2} \lambda_z \left(1 + \frac{\sigma C_{2\alpha}}{8 \mu_d}\right) + \left(\theta_0 + \frac{3}{2} \lambda_z\right) \frac{\sigma C_{2\alpha}}{8 \mu_d}}{\left(1 + \frac{\sigma C_{2\alpha}}{8 \mu_d}\right)^2} \right] \cdot \frac{1}{\left(1 + \frac{\sigma C_{2\alpha}}{8 \mu_d}\right)} \\ &= \frac{1}{6} \rho S \sigma (\Omega R)^2 C_{2\alpha} \cdot \frac{V_x}{V^2} \left[\frac{\frac{3}{2} \lambda_z}{\left(1 + \frac{\sigma C_{2\alpha}}{8 \mu_d}\right)} + \left(\theta_0 + \frac{3}{2} \lambda_z\right) \frac{\frac{\sigma C_{2\alpha}}{8 \mu_d}}{\left(1 + \frac{\sigma C_{2\alpha}}{8 \mu_d}\right)^2} \right] \end{aligned}$$

$$\frac{\partial F_n}{\partial V_z} = \frac{1}{6} \rho S \sigma (\Omega R)^2 C_{2\alpha} \cdot \frac{V_z}{V^2} \left[\frac{\frac{3}{2} \lambda_z}{\left(1 + \frac{\sigma C_{2\alpha}}{8 \mu_d}\right)} + \left(\theta_0 + \frac{3}{2} \lambda_z\right) \frac{\frac{\sigma C_{2\alpha}}{8 \mu_d}}{\left(1 + \frac{\sigma C_{2\alpha}}{8 \mu_d}\right)^2} \right]$$

$$\begin{aligned} \frac{\partial F_n}{\partial \Omega} &= \frac{1}{6} \rho S \sigma (\Omega R)^2 C_{2\alpha} \cdot \frac{2}{\Omega} \cdot \left(\theta_0 + \frac{3}{2} \lambda_z\right) \cdot \frac{1}{\left(1 + \frac{\sigma C_{2\alpha}}{8 \mu_d}\right)} + \\ &\quad \frac{1}{6} \rho S \sigma (\Omega R)^2 C_{2\alpha} \cdot \frac{1}{\Omega} \left[\frac{-\frac{3}{2} \lambda_z}{\left(1 + \frac{\sigma C_{2\alpha}}{8 \mu_d}\right)} - \left(\theta_0 + \frac{3}{2} \lambda_z\right) \frac{\frac{\sigma C_{2\alpha}}{8 \mu_d}}{\left(1 + \frac{\sigma C_{2\alpha}}{8 \mu_d}\right)^2} \right] \\ &= \frac{1}{6} \rho S \sigma (\Omega R)^2 C_{2\alpha} \cdot \frac{1}{\Omega} \cdot \left[\frac{2\theta_0 + \frac{3}{2} \lambda_z}{\left(1 + \frac{\sigma C_{2\alpha}}{8 \mu_d}\right)} - \left(\theta_0 + \frac{3}{2} \lambda_z\right) \frac{\frac{\sigma C_{2\alpha}}{8 \mu_d}}{\left(1 + \frac{\sigma C_{2\alpha}}{8 \mu_d}\right)^2} \right] \end{aligned}$$

Annexe 6 : Commandes en vol équilibré

Pour le vol équilibré nous avons

$$\begin{aligned}\dot{V}_x &= 0 \\ \dot{V}_h &= 0\end{aligned}$$

soit

$$\begin{aligned}-F_n \sin \theta - \frac{1}{2} \rho (SC_x)_f V V_x &= 0 \\ F_n \cos \theta - \frac{1}{2} \rho (SC_x)_f V V_z - M.g &= 0\end{aligned}$$

or en vol horizontal, nous avons $V_z = 0$ donc

$$\begin{aligned}\theta &= \alpha_d \\ -F_n \sin \alpha_d - \frac{1}{2} \rho (SC_x)_f V V_x &= 0 \\ F_n \cos \alpha_d - M.g &= 0\end{aligned}$$

L'incidence du disque rotor sera donnée par :

$$\operatorname{tg} \alpha_d = - \frac{\frac{1}{2} \rho (SC_x)_f V V_x}{M.g}$$



Le pas collectif sera calculé par:

$$F_n = \frac{M.g}{\cos \alpha_d}$$

avec

$$\begin{aligned}F_n &= \frac{1}{6} \rho S \sigma (\Omega R)^2 C_{zm} \\ C_{zm} &= \frac{KC_{zm0}}{\left(1 - KC_{zm1} \frac{\sigma}{12 \mu_d}\right)}\end{aligned}$$

soit

$$\theta_o = \frac{\left(\frac{M.g}{\cos \alpha_d \frac{1}{6} \rho S \sigma (\Omega R)^2} \left(1 - KC_{zm1} \frac{\sigma}{12 \mu_d}\right) + KC_{zm1} \lambda_z \right)}{3C_{z\alpha} \left(\frac{1}{3} + \frac{\mu_d^2}{2} - \frac{\frac{4}{3} \mu_d^2}{\left(1 + \frac{3}{2} \mu_d^2\right)} \right)}$$

NOM DU CANDIDAT:

My-Anh DAO-GARNOT

JURY

Président: P. BORNE	Professeur (EC LILLE - Villeneuve d'Ascq)
Rapporteur: M. KINNAERT	Professeur (U.L.B. - Bruxelles)
Examineurs: C. BRIE	Professeur (I.N.S.A. - Rennes)
G. DAUPHIN-TANGUY	Professeur (EC LILLE - Villeneuve d'Ascq)
H.T. HUYNH	Chef de Subdivision (O.N.E.R.A. - Salon de Provence)
M. STAROSWIECKI	Professeur (E.U.D.I.L. - Lille I)

TITRE DE LA THESE:

*'Contribution au calcul de trajectoires optimales par la méthode de quasilinearisation.
Mise en œuvre et application en aéronautique'*

RESUME

Ce travail, effectué à l'O.N.E.R.A., présente une synthèse de méthodes de calcul de commande optimale pour des systèmes déterministes représentés par un système d'équations différentielles non linéaires, et l'étude détaillée d'une de ces méthodes: la quasilinearisation.

Cette méthode permet de résoudre la plupart des problèmes d'optimisation de trajectoires rencontrés dans le domaine aérospatial, avec la possibilité de prendre en compte des contraintes non linéaires de natures diverses portant à la fois sur les commandes (limitation des amplitudes) et sur l'état (limitation relatives au domaine de vol). Elle présente, par rapport à des méthodes numériques du premier ordre, l'avantage d'une convergence finale quadratique et donc la possibilité d'une utilisation en temps réel.

Néanmoins, comme toutes les méthodes du second ordre, sa limitation demeure la grande sensibilité à l'initialisation. La contribution essentielle de cette thèse est de proposer des modifications permettant d'accroître les possibilités d'utilisation de cette méthode, notamment en augmentant son domaine de convergence et en proposant plusieurs modes d'initialisation. Un programme général est mis en œuvre pour tester l'algorithme.

Ce programme est appliqué dans un premier temps, à des exemples simples, dans un second temps, à deux exemples du domaine aéronautique: le calcul d'une trajectoire d'interception pour un avion de combat et l'optimisation d'une manœuvre d'évitement d'obstacle pour un hélicoptère.

Soutenance prévue le:

9 juin 1992

à 10 heures 30

Bâtiment B - ÉCOLE CENTRALE DE LILLE

Amphithéâtre BODA